

# Area-Efficient Drawings of Outer-1-Planar Graphs

by

Pavle Bulatovic

A thesis  
presented to the University of Waterloo  
in fulfillment of the  
thesis requirement for the degree of  
Master of Mathematics  
in  
Computer Science

Waterloo, Ontario, Canada, 2020

© Pavle Bulatovic 2020

## **Author's Declaration**

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

## Statement of Contributions

I would like to acknowledge the names of my co-authors who contributed to the research described in this dissertation, these include:

- my supervisor, Prof. Therese Biedl

## Abstract

We study area-efficient drawings of planar graphs: embeddings of graphs on an integer grid so that the bounding box of the drawing is minimized. Our focus is on the class of outer-1-planar graphs: the family of planar graphs that can be drawn on the plane with all vertices on the outer-face so that each edge is crossed at most once. We first present two straight-line drawing algorithms that yield small-area straight-line drawings for the subclass of complete outer-1-planar graphs. Further, we give an algorithm that produces an orthogonal drawing of any outer-1-plane graph in  $O(n \log n)$  area while keeping the number of bends per edge relatively small.

## **Acknowledgements**

I would like to thank my advisor Therese Biedl for her expert guidance and advice during my Masters. I also thank my readers Anna Lubiw and Eric Blais for their comments and careful reading of my thesis.

I would also like to thank my friends and family for their continuous support.

# Table of Contents

List of Figures	viii
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Our Results . . . . .	3
1.3 Outline of the Thesis . . . . .	5
<b>2 Background and Preliminaries</b>	<b>6</b>
2.1 Graph Theory . . . . .	6
2.2 Outer-1-Planar Graphs . . . . .	8
2.3 Graph Drawing . . . . .	10
<b>3 Straight-line Drawings of Outer-1-Plane Graphs</b>	<b>12</b>
3.1 Review of Existing Results . . . . .	12
3.2 Extension to Outer-1-Plane Graphs . . . . .	14
3.3 Drawing o1p Graphs in $O(n \cdot \text{depth}(G))$ Area . . . . .	15
3.4 Drawing Complete Outer-1-Plane Graphs . . . . .	19
3.4.1 Review of Di Battista/Frati [15] . . . . .	20
3.4.2 Generalization to Complete Outer-1-Plane Graphs . . . . .	24

<b>4</b>	<b>Orthogonal Drawings of Outer-1-Plane Graphs</b>	<b>47</b>
4.1	Review of Existing Results . . . . .	47
4.2	Review of Biedl’s Algorithm [5] . . . . .	49
4.3	Inserting the Missing Diagonals . . . . .	53
4.4	Conversion to Orthogonal Point-Drawings . . . . .	58
<b>5</b>	<b>Conclusion and Open Problems</b>	<b>64</b>
5.1	Discussion: Straight-line Drawings . . . . .	64
5.2	Discussion: Orthogonal Drawings . . . . .	65
	<b>References</b>	<b>66</b>

# List of Figures

1.1	Left: a maximal outer-1-plane graph $G$ . Right: $skel(G)$ and its weak dual ternary tree. . . . .	3
2.1	An example of a complete outer-planar graph and its weak dual complete binary tree. . . . .	8
2.2	An example of a maximal outer-1-plane graph $G$ , its $skel(G)$ and a <i>half-skel</i> ( $G$ ). . . . .	9
2.3	From left to right: straight-line drawing, poly-line drawing, orthogonal point-drawing and an flat orthogonal box-drawing of the same graph. . . . .	10
3.1	An illustration of the proof of the Theorem 10. . . . .	14
3.2	Final drawing of an outer-1-plane graph with 40 vertices and depth 5. Observe that the drawing occupies 6 rows and 40 columns. . . . .	16
3.3	Placement of face $f$ and recursive step in the proof of Theorem 11. . . . .	17
3.4	Left: An example of a star-shaped drawing of a binary tree rooted at $w$ . Observe that $(SS1)$ and $(SS2)$ hold for polygons $P_l(w)$ and $P_r(u)$ and that $p_s$ and $p_t$ satisfy condition $(SS3)$ . Right: Illustration of the bijection $\gamma$ , observe that if $(f_1, f_2)$ is an edge of the dual tree $T$ , then $(\gamma(f_1), \gamma(f_2))$ is an edge of $G$ . . . . .	21
3.5	Output of Di Battista/Fratti subroutine for drawing a star-shaped binary tree of height three. . . . .	24
3.6	An example of a zigzag star-shaped drawing of a binary tree rooted at node $w_1$ (dark edges). Observe that the left-right path $v_1, \dots, v_6$ of $v_1$ satisfies condition $(ZZS2)$ and that the rightmost path $w_1, \dots, w_6$ satisfies condition $(ZZS3)$ . . . . .	26

3.7	An example of $M(G)$ on a complete o1p graph $G_{st}$ . The dual tree of $M(G)$ rooted at $\rho$ is shown with thick red lines. . . . .	28
3.8	Proof of Lemma 3, face $f$ is the dashed region. . . . .	30
3.9	From left to right: HV, VH and HH drawings and their enclosing triangles. . . . .	32
3.10	Left: An example of non-extreme HV drawing. Right: Illustration that for extreme HV drawings $ y(t(Enc(\Gamma)) - y(\rho) $ is the number of horizontal edges on the leftmost path. . . . .	34
3.11	Left: Illustration of the four regions from Lemma 6 for an extreme $HV$ drawing. Point $q$ is an example of property (V1) and point $p$ is an example of property (V4). Right: An example of a non-extreme HV drawing that does not satisfy property (V4). . . . .	35
3.12	An example of the left lift operation. . . . .	36
3.13	Left: placement of $E_{ll}, E_{lr}, E_{rl}, E_{rr}$ before the lift on $\Gamma_{ll}$ was performed. Right: the final drawing of $\Gamma$ and $Enc(\Gamma)$ (dashed). . . . .	39
3.14	Left: Illustration that $\Gamma$ is star-shaped. Right: Illustration of property (ZZS2) for $\rho$ and $\rho$ .left. Note that the lengths of the edges are not to scale. . . . .	41
3.15	Illustration of the grid size analysis. . . . .	42
3.16	Illustration that vertex $\rho$ still sees the vertices of $L(E_{rl})$ after the left lift. . . . .	43
3.17	Placement of the sub-tree drawings for each invariant before the lifts are performed. In Invariant 2, we perform the right lift on $\Gamma_{rr}$ , the HH drawing. In Invariant 3, we perform the left lift on $\Gamma_{ll}$ and the right lift on $\Gamma_{rr}$ . . . . .	44
3.18	Demonstration that $HV$ drawings satisfy property (ZZS3). . . . .	45
4.1	(Argyriou et al. [2]) (Left) An example of a biconnected outer-1-plane graph which does <i>not</i> admit an embedding-preserving orthogonal drawing so that every edge has at most one bend. (Right) A non-embedding-preserving drawing of the same graph with at most one bend per edge; the middle crossing is not preserved. . . . .	48
4.2	Left: Illustration of $G_1$ and $G_2$ . Right: Further splitting $G_1$ in Case 2. . . . .	50
4.3	Illustration of Case 1. . . . .	50
4.4	Releasing vertex $x$ so it spans the bottom row of the drawing. . . . .	51
4.5	Illustration of combining drawings $\Gamma'_a, \Gamma'_b$ and $\Gamma_2$ in Case 2. . . . .	51

4.6	Case 1 and Case 2 when $ G_1  \geq  G_2 $ . . . . .	52
4.7	Demonstration of Case 1. . . . .	54
4.8	Demonstration of Case 2. . . . .	56
4.9	Illustration that the row above and the row below the horizontal line segment of $(u, w)$ contain no horizontal line segments below $\Gamma'_b$ and $\Gamma'_a$ . . . . .	57
4.10	An example of replacing a box with the corresponding orthogonal configuration. . . . .	58
4.11	Exhaustive list of the box-to-orthogonal conversion rules (up to symmetry). Similar to Argyriou et al. [2]. . . . .	60
4.12	Illustration of the proof of Lemma 7. . . . .	61
4.13	Exhaustive list of replacement rules (up to symmetry) which ensures that every critical incident edge-segment gets at most one extra bend. A similar figure was used in [2]. . . . .	63

# Chapter 1

## Introduction

### 1.1 Motivation

Graph drawing is the study of producing aesthetically pleasing drawings of graphs. It has proven to be useful in a variety of domains such as VLSI design, software diagram visualization, biology and many others [4]. By requiring every vertex to be placed on the integer grid, one can define the area of a drawing to be the number of grid points in the smallest axis-aligned rectangle containing the drawing.

It is natural to attempt to draw graphs without crossings whenever possible. We therefore commonly restrict ourselves to the class of planar graphs, the family of graphs that admit crossing-free embeddings in the plane. Although the very definition of planar graphs allows the edges to be drawn as “curves”, one of the first results in the area, independently proven by Wagner [39], Fary [20] and Stein [34], says that any planar graph can be drawn in the plane crossing-free, so that all its edges are straight-line segments (a *straight-line drawing*). When re-scaled to the integer grid, the area of the drawings obtained from their proofs is unfortunately exponential in the size of the vertex set.

Classical results of the field, independently shown by Schnyder [33] and Fraysseix, Pach and Pollack [13], say that any  $n$ -vertex planar graph admits a planar straight-line drawing in  $O(n^2)$  area. These algorithms are in fact optimal in the sense that there exist families of planar graphs that require  $\Omega(n^2)$  area in any planar straight-line grid drawing [12]. It is therefore natural to ask whether there are any interesting sub-classes of planar graphs that admit planar straight-line drawings in sub-quadratic area. Such drawings have been found for binary trees [32] and outer-planar graphs [15] for instance. In this thesis, we explore area-efficient algorithms for drawing *outer-1-planar* graphs.

Outer-1-planar graphs arise from two well-studied families of graphs: 1-planar graphs and outer-planar graphs. *1-planar* graphs are graphs that can be embedded in the plane so that every edge is crossed at most once. Unlike planar graphs, which can be recognized in linear time [28], recognizing 1-planar graphs is NP-hard [30]. Although not every 1-planar graph admits a 1-planar straight-line drawing, Thomassen [37] and Hong et al. [27] have independently given a characterization of 1-planar graphs that do admit straight-line 1-planar drawings. Both graph-theoretic and graph-drawing results for 1-planar graphs are numerous and we refer the reader to a survey on 1-planar graphs [29] and more generally beyond-planar graphs [16].

Outer-planar graphs are graphs that can be drawn planar so that all vertices are on the outer-face. They can be characterized by forbidding minors  $K_{2,3}$  and  $K_4$ , and recognized in linear time [31]. An *orthogonal point-drawing* is a grid drawing where each edge is a sequence of horizontal or vertical line segments. Observe that for a graph to admit an orthogonal point-drawing, its maximum degree must be at most four. We discuss orthogonal-point drawings in detail in Chapter 4. One of the first graph drawing results on outer-planar graphs dates back to 1981 when Dolev and Trickey [17] showed that every outer-planar graph with degree at most four admits an orthogonal point-drawing in linear area. We review these graph drawing models and other terms used below in detail in Chapter 2. In an attempt to give a straight-line drawing of sub-quadratic area, Garg and Rusu [24] showed that any outer-planar can be drawn straight-line in  $O(dn^{1.48})$  area, where  $d$  is the maximum degree of the graph. This bound was later improved to  $O(n^{1.48})$  by DiBattista and Frati [15] and to  $O(dn \log n)$  by Frati [22]. The most recent result by Frati, Patrignani and Roselli [23] says that every outer-planar graph admits a straight-line drawing in  $O(n^{1+\epsilon})$  area for any  $\epsilon > 0$ . Biedl gave an algorithm that produces a flat orthogonal box-drawing, and hence a poly-line drawing, of any outer-planar graph in  $O(n \log n)$  area [5]. She has further shown in [6] that every outer-planar graph  $G$  admits a flat orthogonal box-drawing in  $O(n \cdot pw(G))$  area, where  $pw(G)$  is the pathwidth of  $G$ .

Of special interest to us is the combination of 1-planar and outer-planar graphs. An *outer-1-planar (o1p)* graph is a graph which can be drawn in the plane so that all of its vertices are on the outer-face and every edge is crossed at most once. They were first introduced and studied by Eggleton in 1984 [18], where he proved some of their basic properties, for example maximal outer-1-plane graphs have no crossings on the outer-face. Unlike outer-planar graphs, o1p graphs cannot be characterized by the exclusion of graph minors [3]. Still, it was independently shown by Auer et al. [3] and Hong et al. [26] that outer-1-planarity can be tested in linear time. Both algorithms are based on the decomposition of outer-1-planar graphs using SPQR-trees and output the final o1p embedding if it exists. Auer et al. [3] have further shown that o1p graphs have treewidth

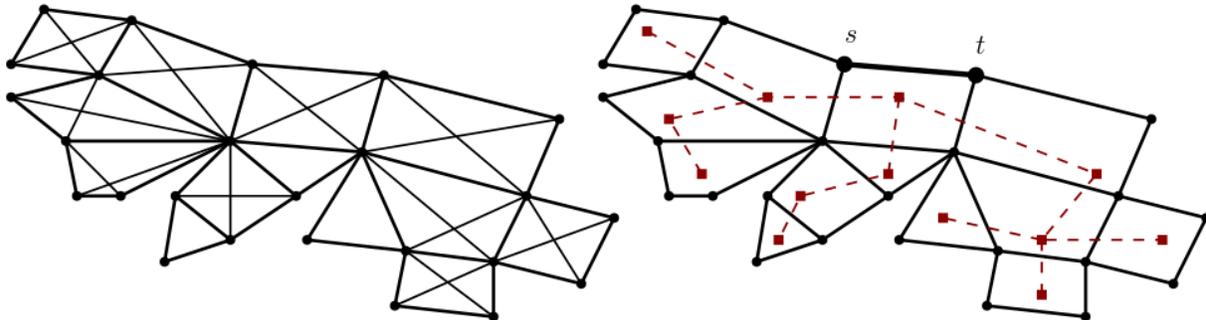


Figure 1.1: Left: a maximal outer-1-plane graph  $G$ . Right:  $skel(G)$  and its weak dual ternary tree.

at most three, stack number at most two, queue number at most three and that every o1p graph is planar. Multiple researchers have investigated graph drawings of o1p graphs. It was first shown by Dehkordi and Eades [14] that all o1p graphs admit RAC drawings, i.e., straight-line drawings so that every crossing is at the right angle. Further, every o1p graph admits a straight-line drawing in  $O(n^2)$  area such that all the vertices are on the outer-face and a (not embedding-preserving) flat orthogonal box-drawing in  $O(n \log n)$  area [3]. We discuss *embedding-preserving* drawings in Chapter 2. Di Giacomo, Liotta and Montecchiani have shown that every o1p graph can be drawn straight-line using at most  $O(d)$  slopes, where  $d$  is the maximum degree of a graph. Most recently it was shown by Argyriou et al. [2] that every biconnected o1p graph admits an orthogonal point-drawing in  $O(n^2)$  area so that every edge has at most two bends. We will give a more detailed review of the relevant graph drawing results in Chapter 3 and Chapter 4.

## 1.2 Our Results

An *outer-1-plane* graph is an outer-1-planar graph with a given outer-1-planar embedding. In the first part of the thesis we focus on the outer-1-plane graphs of small “depth” and on “complete” outer-1-plane graphs, and give area-efficient straight-line drawings for these two sub-classes of outer-1-plane graphs. We say an outer-1-plane graph is *maximal* if no edge can be added while staying simple and outer-1-plane. To define notions of “depth” and “complete” outer-1-plane graphs, we associate every *maximal* outer-1-plane graph  $G$  with its *skeleton*,  $skel(G)$ . Namely, to obtain  $skel(G)$  simply remove all crossings from  $G$ , see Figure 1.1.

It is easy to see that the dual tree of  $skel(G)$  is always a *ternary tree*, after deleting the vertex corresponding to the outer-face. We assume that this tree is rooted after selecting some distinguished reference edge  $(s, t)$  of  $skel(G)$  on the outer-face, and choosing the *root* face to be the interior face of  $skel(G)$  containing the edge  $(s, t)$ . Hence, a natural parameter of any maximal o1p graph is  $depth(G)$ , i.e, the number of nodes on the longest root-to-leaf path of the dual tree of  $skel(G)$ . The reference edge  $(s, t)$  of  $skel(G)$  is chosen so that  $depth(G)$  is minimized. For a (non-maximal) outer-1-plane graph  $G$ , we use  $depth(G)$  to denote the smallest depth among all maximal outer-1-plane graphs containing  $G$  (with the same number of vertices as  $G$ ). We begin the thesis by giving a very simple algorithm to draw any o1p graph so that the height of the drawing is proportional to  $depth(G)$ :

**Theorem 1** *Let  $G$  be an  $n$ -vertex outer-1-plane graph. Then  $G$  admits an embedding-preserving straight-line outer-1-plane drawing in a grid of height  $O(depth(G))$  and width  $O(n)$ .*

In the subsequent section, we introduce *complete* outer-1-plane graphs. First, a *complete* ternary tree is a ternary tree in which every “level” must be completely filled. So, we say a maximal o1p graph  $G$  is *complete* if its rooted dual tree of  $skel(G)$  is a complete ternary tree. Similarly, a (non-maximal) outer-1-plane graph is complete if it can be augmented to a complete maximal o1p graph without adding vertices. We prove the following theorem:

**Theorem 2** *Let  $G$  be an  $n$ -vertex complete outer-1-plane graph. Then  $G$  admits an embedding-preserving straight-line outer-1-plane drawing of width and height  $O(n^{0.63})$ .*

This result was motivated by the linear-area straight-line drawings of *complete* outer-planar graphs by DiBattista and Frati [15]. Observe that complete outer-1-plane graphs have depth  $O(\log n)$ , hence Theorem 1 would yield  $O(n \log n)$  area drawings. However, Theorem 2 gives drawings with better *aspect ratio*, as both the width and the height are sub-linear.

We now move to orthogonal point-drawings of outer-1-plane graphs. The main result of this chapter is the following theorem:

**Theorem 3** *Let  $G$  be an  $n$ -vertex outer-1-plane graph of maximum degree 4. Then  $G$  admits an embedding-preserving orthogonal point-drawing in  $O(n \log n)$  area such that every edge has at most seven bends.*

This theorem was inspired by the result of Argyriou et al. [2], who showed that any 1-plane graph (and hence outer-1-plane) admits an embedding-preserving orthogonal point-drawing in  $O(n^2)$  area such that every edge has at most three bends. For *biconnected* outer-1-plane graphs, they reduced the number of bends per edge to two. This result is optimal in the sense that there exists an o1p graph  $G$  so that in any embedding-preserving orthogonal point-drawing of  $G$ , there must be at least one edge that has at least two bends [2]. Our aim is to produce orthogonal point-drawings of o1p graphs in sub-quadratic area while keeping the number of bends per edge relatively small.

## 1.3 Outline of the Thesis

The remainder of the thesis is organized as follows:

- In Chapter 2 we introduce standard definitions and notations from graph theory and graph drawing, including the most relevant families of graphs such as outer-planar, 1-planar and outer-1-planar graphs.
- In Chapter 3 we present small-area straight-line drawing algorithms for outer-1-plane graphs with small depth and for complete outer-1-plane graphs. We begin the chapter by reviewing the existing straight-line drawing algorithms for both outer-planar and outer-1-planar graphs and presenting the  $O(n \cdot \text{depth}(G))$  area approach (Theorem 1). Then, after reviewing the techniques used by DiBattista and Frati of [15], we prove Theorem 2.
- In Chapter 4 we give small-area orthogonal drawings of outer-1-plane graphs while keeping the number of bends per edge relatively small. We first review the relevant orthogonal graph drawing algorithms and motivate our result. Then, after giving a detailed review of Biedl’s algorithm [5] for producing flat orthogonal box-drawings of outer-planar graphs, we extend the technique to the super-class of outer-1-plane graphs. Finally, we conclude the chapter with a proof of Theorem 3.
- In Chapter 5 we conclude with several open problems.

# Chapter 2

## Background and Preliminaries

### 2.1 Graph Theory

A *graph*  $G = (V, E)$  is a pair of two sets  $V$  and  $E$ , which we refer to as *vertices* and *edges*, respectively. An edge is a pair  $(u, v)$  of two vertices, so a graph is most commonly visualized as collection of dots in the plane (vertices) that are connected by curves (edges). We assume that no two vertices are connected by two distinct edges and that no vertex is related to itself by an edge; in general such graphs are called *simple* graphs. We use  $V(G)$  and  $E(G)$  to denote the vertex and the edge set of a graph  $G$ , respectively. We usually use  $|G|$  to denote the size of the vertex set of  $G$ . All graphs in this thesis are *undirected*, that is, the order in which the endpoints  $u, v$  of an edge  $(u, v)$  are listed is irrelevant.

If  $(u, v)$  is an edge, we say that vertices  $u$  and  $v$  are *adjacent* or *neighbors*; we also say that edge  $(u, v)$  is *incident* to both  $u$  and  $v$ . The number of neighbors of a vertex  $v$  is called the *degree* of  $v$ , which we denote by  $\text{deg}(v)$ . A path in a graph is a sequence of distinct vertices  $v_1, v_2, \dots, v_k$  so that for all  $i \in \{1, \dots, k - 1\}$ ,  $(v_i, v_{i+1})$  is an edge. We say that a graph is *connected* if for all  $v, u \in V(G)$ , there exists a path from  $v$  to  $u$ . The notion of connectedness can be generalized to *k-connected graphs*, i.e, graphs that stay connected after fewer than  $k$  vertices and their incident edges are removed. A *cycle* is a path  $p_1, \dots, p_k$  such that  $(p_k, p_1)$  is an edge.

A *tree* is a connected graph without cycles. A rooted binary tree is a tree with a designed root vertex that is allowed to have at most two neighbors, and every other vertex has at most three neighbors. For every edge  $(u, v)$  such that  $u$  is closer to the root than  $v$ , we say that  $v$  is a *child* of  $u$ , and that  $u$  is a *parent* of  $v$ . Once the ordering of the children

is fixed for each vertex  $v$ , we use  $v.left$  and  $v.right$  to denote the left and the right child of  $v$ . Analogous to a binary tree is a *ternary tree*, i.e., a rooted tree such that all of its vertices have at most three children. Non-root nodes with degree one are called the *leaves* of the tree. The *height* of a rooted binary (ternary) tree  $T$  is the number of edges on its longest root-to-leaf path. A *complete binary (ternary) tree* is a rooted binary (ternary) tree in which all non-leaf nodes have the maximum number of children and all root-to-leaf paths have the same length.

A graph  $G$  is *planar* if it can be drawn in the plane without creating any crossings. A specific drawing without crossings of  $G$  is called a *planar embedding* of  $G$ . It is important to note that a planar graph may have more than one planar embedding. Once the embedding is fixed, the edges decompose the plane graph into connected regions that we call *faces*. The unbounded region is called the *outer-face* and all the remaining, closed faces are called the *interior faces*. We say two faces are *adjacent* if they have an edge in common. We say two planar embeddings are *equivalent* if for each vertex  $v$ , the cyclic order of the edges incident to  $v$  is equal in both embeddings and both embeddings have the same outer-face. A *plane graph* on the other hand is a planar graph with a fixed planar embedding. With every plane graph  $G$  we consider another plane graph  $G^*$ , the *strong dual* of  $G$ , in the following way: The set of faces of  $G$  will form the vertex set of  $G^*$ , and if two faces  $f_1$  and  $f_2$  are adjacent, then add  $(f_1, f_2)$  to the edge set of  $G^*$ . By removing the dual vertex of the outer-face and its incident edges, we obtain the *weak dual* of  $G$ . We say  $G$  is a *maximal planar* graph if adding any edge to  $G$  would make it not planar, or not simple. Observe that every face of a maximal plane graph is a triangle, i.e., a cycle of three vertices, hence they are commonly called *triangulated* planar graphs.

An *outer-planar* graph is a graph that can be drawn planar such that all of its vertices are on the outer-face. An *outer-plane* graph is an outer-planar graph with a given outer-planar embedding. A graph is a *maximal outer-planar* if adding any edge to it would make it non-outer-planar. Weak duals of outer-plane graphs are of special importance in this thesis as they are commonly used in producing small area graph drawings. Observe that the weak dual graph of any maximal outer-plane graph is a rooted binary tree, after appropriately selecting the root face. We commonly refer to it as the *dual tree* instead of the “weak dual tree”. We use  $G_{st}$  to denote a maximal outer-plane graph with a given *reference* edge  $(s, t)$  on its outer-face with  $s$  before  $t$  in clockwise order. We commonly refer to the vertices  $s$  and  $t$  as the *poles* of  $G$ . For the given edge  $(s, t)$ , we use  $T_{st}$  to denote the dual binary tree of  $G_{st}$  rooted at the face containing edge  $(s, t)$ . We say  $G_{st}$  is a *complete outer-planar* graph if  $T_{st}$  is a complete binary tree, see Figure 2.1. Without the specified reference edge, we say that maximal outer-plane graph  $G$  is *complete* if there exists an edge  $(s, t)$  on its outer-face such that  $G_{st}$  is complete.

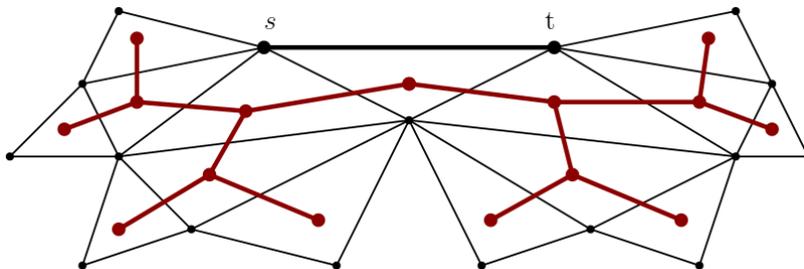


Figure 2.1: An example of a complete outer-planar graph and its weak dual complete binary tree.

We now proceed to *1-planar graphs*, graphs that can be drawn in the plane so that every edge is crossed at most once. Similar to planar graphs, we distinguish between *1-planar* and *1-plane graphs*. To define the equivalence of two 1-planar embeddings, we first need to *planarize* a 1-plane graph. We do this by inserting a *dummy* vertex for each crossing in the given 1-planar embedding  $\Gamma$ , to obtain a plane graph  $\Gamma^P$ , which we call the *planarization* of  $\Gamma$ . Now, we say two 1-planar embeddings  $\Gamma_1$  and  $\Gamma_2$  are *equivalent*, if  $\Gamma_1^P$  and  $\Gamma_2^P$  are equivalent as planar embeddings. Note that exactly the same pairs of edges cross in two equivalent drawings. Equivalence of two embeddings will be important for defining *embedding-preserving* graph drawings below.

## 2.2 Outer-1-Planar Graphs

The combination of outer-planar and 1-planar graphs gives rise to *outer-1-planar* (o1p) graphs; graphs which can be embedded in the plane so that all the vertices are placed on the outer face and each edge is crossed at most once. See Figure 2.2 (left) for an example. We say that an outer-1-planar graph  $G$  is *maximal* if adding any edge would make  $G$  not outer-1-planar or not simple. We commonly state graph drawing results for maximal outer-1-plane graphs only since any non-maximal o1p graph can be easily augmented to a maximal one by a simple procedure described by Dekhordi and Eades in [14]. Then, by simply removing the augmented edges from a maximal o1p graph, we obtain a drawing of the original graph.

We now state an observation proven in [14]:

**Observation 1** (Dekhordi and Eades [14]) *Suppose that  $G$  is a maximal outer-1-plane graph and edges  $(a, c)$  and  $(b, d)$  cross. Then  $\{a, b, c, d\}$  induces a complete subgraph of  $G$ .*

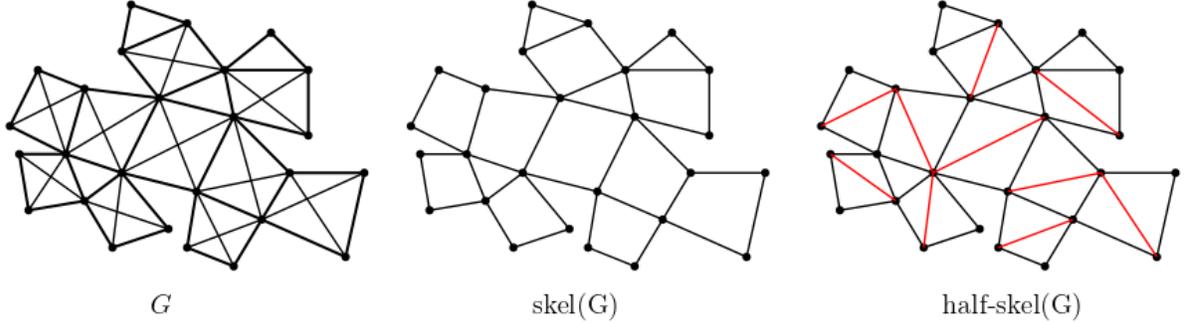


Figure 2.2: An example of a maximal outer-1-plane graph  $G$ , its  $skel(G)$  and a  $half-skel(G)$ .

Further, edges  $(a, b)$ ,  $(b, c)$ ,  $(c, d)$  and  $(d, a)$  have no crossing.

Let  $G$  be a maximal o1p graph. By removing all pairs of crossing edges from  $G$  we obtain another outer-1-plane graph that we denote by  $skel(G)$ , the *skeleton* of  $G$ , see Figure 2.2. Skeletons have a very simple structure that will be convenient to us throughout the thesis. Using the above observation, Dekhordi and Eades prove the following:

**Lemma 1** (Dekhordi and Eades [14]) *If  $G$  is a maximal outer-1-plane graph, then  $skel(G)$  is an outer-plane graph in which every inner face is either a 3-cycle or a 4-cycle.*

We refer to the faces of  $skel(G)$  as *quadrangles* and *triangles*. We call the edges of  $G \setminus skel(G)$  the *missing diagonals* of  $skel(G)$ . Now, by inserting back one (of the two) missing diagonals in each quadrangle of  $skel(G)$ , we obtain a maximal outer-plane graph; we call this a *half-skeleton* of  $G$ , and denote it by  $half-skel(G)$ . Note that  $skel(G)$  has multiple half-skeletons, depending on which missing diagonals we insert. See also Figure 2.2 for an example.

Similar to outer-plane graphs, we use  $G_{st}$  to denote an o1p graph with specified reference edge  $(s, t)$  on the outer-face of  $skel(G)$ , and  $T_{st}$  to denote the dual ternary tree of  $skel(G)$  rooted at the interior face of  $skel(G)$  containing the edge  $(s, t)$ .

Further, we define  $depth(G_{st})$  to be the number of vertices on the longest root-to-leaf path of  $T_{st}$ . If  $G_{st} = (s, t)$ , then we set  $depth(G_{st})$  to be zero. If the reference edge  $(s, t)$  is not specified, then  $depth(G) = \min\{depth(G_{st}) \mid (s, t) \text{ is an edge on the outer-face of } G\}$ . For a single vertex graph  $G$ , we also set  $depth(G)$  to be zero.

Finally, we say that  $G_{st}$  is *complete* if the dual ternary tree of  $skel(G)$  rooted at the face containing edge  $(s, t)$  is a complete ternary tree. Analogous to complete outer-plane graphs,

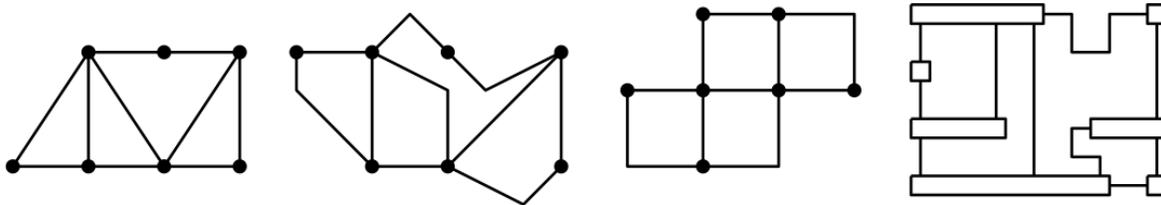


Figure 2.3: From left to right: straight-line drawing, poly-line drawing, orthogonal point-drawing and an flat orthogonal box-drawing of the same graph.

we say that an o1p graph  $G$  is *complete* if there exists an edge  $(a, b)$  on the outer-face of  $skel(G)$  so that  $G_{ab}$  is complete.

## 2.3 Graph Drawing

Recall that a *graph drawing* is an assignment of a geometric shape such as a point to each vertex. Unless specified otherwise, vertices are drawn as points. All graph drawings in this thesis are *grid drawings*, i.e., all vertices must be placed on grid points. The *area* of a drawing  $\Gamma$  is the number of grid points occupied by the smallest axis-aligned rectangle containing  $\Gamma$ . The *height* of  $\Gamma$  is the total number of horizontal lines, or rows, with integer  $y$ -coordinates that are occupied by  $\Gamma$ . Similarly, the *width* of  $\Gamma$  is the total number of vertical lines, or columns, with integer  $x$ -coordinates that are occupied by  $\Gamma$ . We use  $height(\Gamma)$  and  $width(\Gamma)$  to denote the height and the width of  $\Gamma$ . The *aspect ratio* of the drawing is defined as the ratio of the width to the height of the drawing (or height to width, depending on which side is larger). We say that a drawing  $\Gamma$  is *embedding-preserving* for a graph  $G$  along with a given embedding if the embedding implied by  $\Gamma$  is equivalent to the given embedding of  $G$ . In particular, for 1-planar graphs we must preserve the order of edges around each vertex and the set of crossings. Given a grid drawing  $\Gamma$ , we use  $x(v)$  and  $y(v)$  to denote the  $x$  and  $y$  coordinates of the vertex  $v$ . Of special importance in this thesis are the following types of drawings:

- A **straight-line drawing** is a grid drawing in which every edge is drawn as a straight-line segment.
- A **poly-line drawing** is a grid drawing where each edge is represented by a sequence of straight-line segments and each endpoint of a line segment (*bend*) must also be placed on a grid point.

- An **orthogonal point-drawing** is a poly-line drawing where each edge is a sequence of horizontal or vertical line segments. Observe that for a graph to admit an orthogonal point-drawing, its maximum degree must be at most four.
- A **flat orthogonal box-drawing** is a grid drawing where vertices are drawn as horizontal straight-line segments (or elongated boxes) with both of its endpoints placed on the grid, and each edge is a sequence of horizontal or vertical line segments. Each endpoint of a line segment (*bend*) must also be placed on a grid point. This is the only type of drawing in this thesis where vertices are not represented as points. In this thesis, we require each vertex (box) to occupy one row only, we enlarge the boxes in figures for readability purposes only.

See Figure 2.3 for an example of each type of drawing.

# Chapter 3

## Straight-line Drawings of Outer-1-Plane Graphs

In this chapter we show that every outer-1-plane graph admits a straight-line drawing in an  $O(n) \times O(\text{depth}(G))$ -grid. We further show that complete outer-1-plane graphs can be drawn straight-line in an  $O(n^{1.63}) \times O(n^{1.63})$ -grid.

### 3.1 Review of Existing Results

We first review planar straight-line drawing results for the subclass of outer-planar graphs. The first attempt to obtain sub-quadratic area straight line drawings of outer-planar graphs was by Garg and Rusu in 2007 [24]; they proved the following:

**Theorem 4** (*Garg and Rusu [24]*) *Every outer-planar graph  $G$  with maximum degree  $d$  admits an outer-planar straight-line drawing in  $O(dn^{1.48})$  area.*

Their result was further improved by Di Battista and Frati:

**Theorem 5** (*Di Battista and Frati [15]*) *Every outer-planar graph  $G$  admits an outer-planar straight-line drawing in  $O(n^{1.48})$  area.*

The algorithm works by first augmenting  $G$  to a maximal outer-plane graph, obtaining a so-called “star-shaped” drawing of its dual binary tree, and then recovering the original

outer-plane graph from the dual tree drawing. In Section 3.4, we will review this result in detail as the ideas presented in their paper will be needed in this thesis. They further showed that any complete outer-planar graph can be drawn straight-line in linear area:

**Theorem 6** (Di Battista and Frati [15]) *Every complete outer-planar graph  $G$  admits an outer-planar straight-line drawing in  $O(n)$  area.*

Afterwards, Frati presented a novel  $O(dn \log n)$  area algorithm:

**Theorem 7** (Frati [22]) *Every outer-planar graph  $G$  with maximum degree  $d$  admits an outer-planar straight-line drawing in  $O(dn \log n)$  area.*

The most recent result by Frati, Patrignani and Roselli states the following:

**Theorem 8** (Frati, Patrignani and Roselli [23]) *Every outer-planar graph  $G$  admits an outer-planar straight-line drawing in  $O(n \cdot 2^{\sqrt{2 \log n}} \sqrt{\log n})$  area.*

They remark that the area of the drawing is  $o(n^{1+\epsilon})$  for any fixed  $\epsilon > 0$ . Still, the problem of closing the trivial linear lower bound and the  $o(n^{1+\epsilon})$  upper bound remains open.

We now move to the super-class of outer-1-planar graphs. First, since o1p graphs are planar [3], it follows that they do in fact admit planar straight-line drawings in  $O(n^2)$  area. These drawings, however, do not preserve the crossings and may violate outer-planarity. Auer et al. [3] achieved the same area bound while drawing all the vertices on the outer-face:

**Theorem 9** (Auer et al. [3]) *Every outer-1-plane graph  $G$  admits a straight-line outer-1-plane drawing in  $O(n^2)$  area such that all vertices are on the outer-face.*

Although the question of finding sub-quadratic area straight-line drawings of outer-1-plane graphs is still open, researchers have studied straight-line o1p drawings using different aesthetic criteria. For instance, Eades and Dekhordi [14] showed that any o1p graph admits a straight-line drawing so that every crossing is at a right-angle. Also, Di Giacomo, Liotta and Montecchiani [25] showed that any o1p graph can be drawn with few slopes. Namely, any o1p graph of maximum degree  $d$  admits a straight-line o1p drawing using at most  $O(d)$  different slopes.

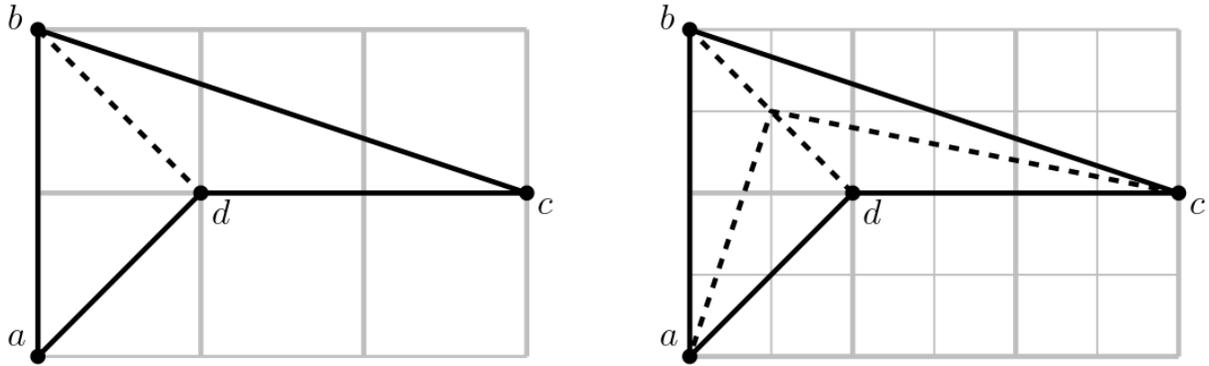


Figure 3.1: An illustration of the proof of the Theorem 10.

### 3.2 Extension to Outer-1-Plane Graphs

Given the many straight-line drawing algorithms for outer-planar graphs, it is natural to ask whether one could draw any o1p graph by a simple reduction from an already existing outer-planar drawing. Let  $G$  be a maximal o1p graph and recall that  $skel(G)$ , i.e., the graph obtained by removing all the crossings of  $G$ , consists of triangular and quadrangular faces only. Observe that if we could draw an outer-plane graph  $skel(G)$  with all quadrangles strictly convex, we could easily recover a drawing of  $G$  by inserting the missing diagonals in each quadrangle as straight-line segments. Unfortunately, none of the already existing results for outer-planar graphs have this property. However, we can easily obtain poly-line drawings with at most one bend per edge of an o1p graph, given an algorithm that draws outer-planar graphs straight-line, by just doubling both the height and width of the drawing:

**Theorem 10** *Let  $G$  be a maximal  $n$ -vertex o1p graph and let  $H$  be a half-skeleton of  $G$ . If  $H$  admits an embedding-preserving planar straight-line drawing in  $f(n)$  area, then  $G$  admits an outer-1-plane, poly-line drawing with at most one bend per edge in  $4f(n)$  area.*

*Proof.* Recall that  $H$  is a maximal outer-plane graph and let  $\Gamma$  be the straight-line drawing of  $H$  in  $f(n)$  area. First, double both the height and the width of the drawing. We now add every missing diagonal with at most one bend. Let  $(a, c)$  be an arbitrary missing diagonal of  $H$ , and let  $f$  be the quadrangle of  $skel(G)$  containing the (missing) edge  $(a, c)$ . Without loss of generality, assume the vertices of  $f$  are  $abcd$  in clockwise order and note that the diagonal  $(b, d)$  is already drawn as a straight-line segment (as it belongs to the

half-skeleton  $H$ ). Let  $p$  be the half-way point of line-segment  $(b, d)$ . Since the coordinates of  $b$  and  $d$  are integers, and since we doubled both the height and the width of  $\Gamma$ , it follows that  $x(p)$  and  $y(p)$  are integers as well. Therefore, simply insert  $(a, c)$  as a sequence of two line segments  $(a, p)$  and  $(p, c)$ .  $\square$

Combining this with Theorem 8 gives:

**Corollary 1** *Every  $n$ -vertex outer-1-plane graph has an outer-1-plane poly-line drawing with at most one bend per edge in  $O(n \cdot 2^{\sqrt{2 \log n}} \sqrt{\log n})$  area.*

Thus we achieve sub-quadratic area drawings if bends are allowed. There does not seem to be an obvious way to tweak straight-line drawings of outer-planar graphs to make each quadrangle strictly convex. Despite repeated attempts, we have not managed to find sub-quadratic area drawings of o1p graphs. In the rest of this chapter we therefore give algorithms which work very well for interesting sub-classes of o1p graphs.

### 3.3 Drawing o1p Graphs in $O(n \cdot \text{depth}(G))$ Area

In this section we give a simple algorithm that produces small area drawings of o1p graphs with small depth. It is important to note that the drawings presented in this section are identical to those given by Auer et al. [3]. The drawing algorithm of Auer et al. is a reduction from a more general approach, due to Alam et al. [1], that draws any 3-connected 1-planar graph in  $O(n^2)$  area, which in turn is a modification of the algorithm by Chrobak and Kant [8]. We present a simple, self-contained drawing algorithm so that the area of the drawing is  $O(n \cdot \text{depth}(G))$  (although in the worst case, the area is quadratic). To achieve this, we use *trapezoidal drawings* that were first introduced by Garg and Rusu [24] for purpose of drawing outer-planar graphs straight-line in  $O(d \cdot n^{1.48})$  area. Drawing outer-1-plane graphs of small depth will motivate the next section where we improve the aspect ratio for the subclass of complete o1p graphs.

Let  $G_{st}$  be a maximal outer-1-plane graph along with some  $(s, t)$  edge on its outer-face, with  $s$  before  $t$  in counter-clockwise order. Recall that  $\text{depth}(G_{st})$  is the number of vertices on the longest root-to-leaf path of  $T_{st}$ , i.e, the dual ternary tree rooted at the interior face containing the edge  $(s, t)$ .

Now, let  $v_1, v_2, \dots, v_n$  be the vertices of  $G_{st}$  sorted in clockwise order around the outer-face, where  $s = v_1$  and  $t = v_n$ . For any edge  $(v_i, v_j)$ , we use  $G_{v_i v_j}$  to denote the subgraph induced by vertices  $v_i, \dots, v_j$ . When  $f_{st} = suvt$  is a quadrangle, the only relevant subgraphs

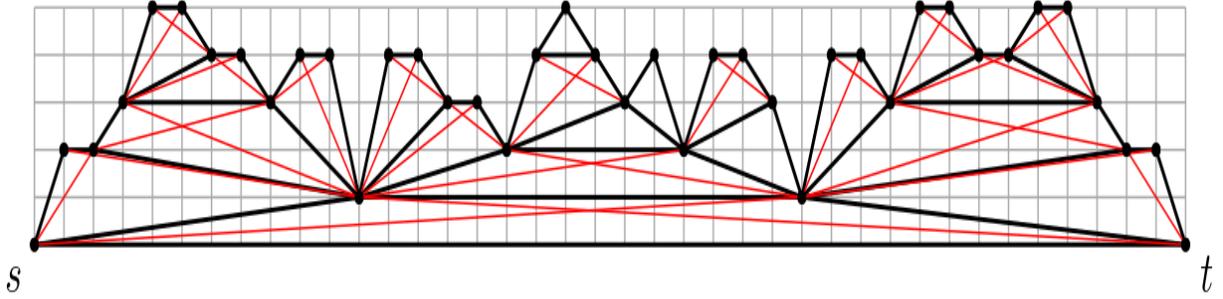


Figure 3.2: Final drawing of an outer-1-plane graph with 40 vertices and depth 5. Observe that the drawing occupies 6 rows and 40 columns.

in this section will be  $G_{su}$ ,  $G_{uv}$  and  $G_{vt}$ ; similarly if  $f_{st}$  is a triangle, there will be two such subgraphs only.

The drawing algorithm is quite straightforward: given a face  $f_{st} := suvt$ , we draw  $f$  in a strictly convex way while making sure that  $(s, u)$ ,  $(u, v)$  and  $(v, t)$  edges are drawn “wide enough” to leave space for the recursive drawings of three child graphs  $G_{su}$ ,  $G_{uv}$ ,  $G_{vt}$ . To achieve small height, we make sure to add exactly one extra row to the parent drawing given the subgraph drawings. Formally, we use the *trapezoidal drawings* that were introduced by Garg and Rusu [24] for outer-planar graph drawings:

**Definition 1** We say  $\Gamma_{st}$  is a trapezoidal drawing of an outer-1-plane graph  $G_{st}$  if:

1.  $\Gamma_{st}$  is an embedding-preserving, outer-1-plane, straight-line grid drawing.
2. For any  $w \in G_{st}$  such that  $w \neq s, t$ , we have  $x(s) < x(w) < x(t)$  and  $y(w) > \max\{y(s), y(t)\}$ .

See Figure 3.2 for an example of a trapezoidal drawing and the output of the drawing algorithm to be presented in Theorem 11. We finally state the graph drawing algorithm:

**Theorem 11** Let  $G_{st}$  be a maximal outer-1-plane graph along with edge  $(s, t)$  on its outer-face. If edge  $(s, t)$  is already drawn such that  $x(t) - x(s) = |G_{st}| - 1$ , then there exists a drawing  $\Gamma_{st}$  of  $G_{st}$  such that:

1.  $\Gamma_{st}$  is trapezoidal.
2.  $height(\Gamma_{st}) = depth(G_{st}) + |y(t) - y(s)| + 1$  and  $width(\Gamma_{st}) = |G_{st}|$ .

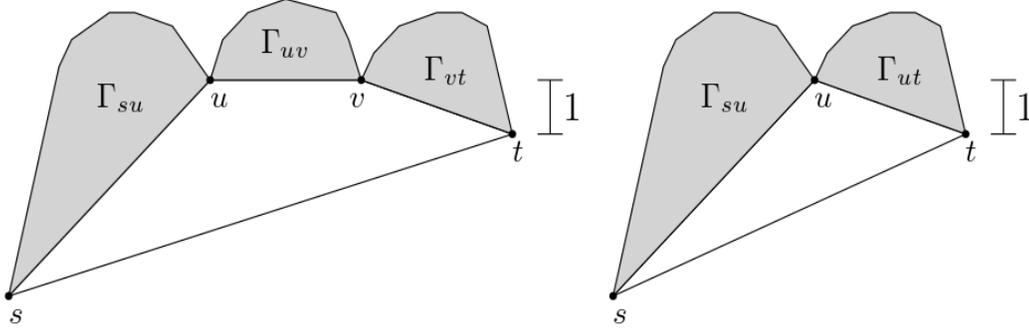


Figure 3.3: Placement of face  $f$  and recursive step in the proof of Theorem 11.

*Proof.* Without loss of generality suppose that  $y(s) \leq y(t)$ . We prove the theorem by induction on  $\text{depth}(G_{st})$ .

*Base Case.* If  $\text{depth}(G_{st}) = 0$ , then  $G_{st} = (s, t)$ , in which case the result follows immediately.

*Recursive Step.* If  $\text{depth}(G_{st}) \geq 1$ , then inductively suppose that the result holds for any o1p graph with depth less than  $\text{depth}(G_{st})$ . Let  $f$  be the interior face of  $\text{skel}(G_{st})$  containing the edge  $(s, t)$ .

*Case 1.*  $f$  is a quadrangle.

Then  $f := suvt$  for some  $u, v \in V(G_{st})$  such that  $u$  comes before  $v$  in clockwise order. We first complete the drawing of  $f$  in the following way:

- Place  $u$  at  $(x(s) + |G_{su}| - 1, y(t) + 1)$ .
- Place  $v$  at  $(x(u) + |G_{uv}| - 1, y(t) + 1)$ .
- Connect  $s$  to  $u$ ,  $u$  to  $v$  and  $v$  to  $t$  by a straight-line segment.

Observe that  $f$  is strictly convex, so the crossing edges  $(s, v)$  and  $(u, t)$  can also be drawn as straight-line segments. Now, since edges  $(s, u)$ ,  $(u, v)$  and  $(v, t)$  are already drawn, we will use them as reference edges for the recursive drawings of  $G_{su}$ ,  $G_{uv}$  and  $G_{vt}$ , respectively. First, observe that by the placement of vertices  $v$  and  $u$ , the condition of the theorem statement is satisfied for the inductive hypothesis of  $G_{su}$  and  $G_{uv}$ . We show that the same

is true for  $G_{vt}$ , namely  $x(t) - x(v) = |G_{vt}| - 1$ . This holds because

$$\begin{aligned}
x(t) - x(v) &= (x(t) - x(s) - (x(v) - x(s))) \\
&= (|G_{st}| - 1) - (x(v) - x(s)) \\
&= ((|G_{su}| + |G_{uv}| + |G_{vt}| - 2) - 1) - ((x(s) + |G_{su}| - 1 + |G_{uv}| - 1) - x(s)) \\
&= |G_{vt}| - 1
\end{aligned}$$

Here the second equality holds because we know  $x(t) - x(s) = |G_{st}| - 1$ . The third equality holds because  $|G_{st}| = |G_{su}| + |G_{uv}| + |G_{vt}| - 2$  ( $u, v$  are counted twice), and by substituting  $x(v)$  by its value, as defined above.

Finally, since  $\text{depth}(G_{su}), \text{depth}(G_{uv}), \text{depth}(G_{vt}) < \text{depth}(G_{st})$ , recursively construct their respective drawings  $\Gamma_{su}, \Gamma_{uv}$  and  $\Gamma_{vt}$ , satisfying properties (1) and (2) of Theorem 11, to obtain the final drawing  $\Gamma_{st}$ . See Figure 3.3 (left) for the demonstration of the recursive step. Since the drawings of all three subgraphs are trapezoidal,  $\Gamma_{su}, \Gamma_{uv}$  and  $\Gamma_{vt}$  cannot have intersections except at the common vertices  $u$  and  $v$ , hence  $\Gamma_{st}$  must be an embedding-preserving, outer-1-plane, straight-line drawing. Observe that the resulting drawing is trapezoidal. Since  $x(t) - x(s) = |G_{st}| - 1$ , by the definition of trapezoidal it immediately follows that the width of  $\Gamma_{st}$  is  $|G_{st}|$ . For the height, we consider all three recursive drawings as follows:

$$\begin{aligned}
\text{height}(\Gamma_{st}) &= \max\{\text{height}(\Gamma_{su}), y(t) - y(s) + 1 + \text{height}(\Gamma_{uv}), y(t) - y(s) + \text{height}(\Gamma_{vt})\} \\
&= \max\{\text{depth}(G_{su}) + y(u) - y(s) + 1, \\
&\quad y(t) - y(s) + 1 + (\text{depth}(G_{uv}) + y(v) - y(u) + 1), \\
&\quad y(t) - y(s) + (\text{depth}(G_{vt}) + y(v) - y(t) + 1)\} \\
&= \max\{\text{depth}(G_{su}) + (y(t) + 1) - y(s) + 1, \\
&\quad \text{depth}(G_{uv}) + y(t) - y(s) + 2, \\
&\quad \text{depth}(G_{vt}) + (y(t) + 1) - y(s) + 1\} \\
&= \text{depth}(G_{st}) + y(t) - y(s) + 1
\end{aligned}$$

Here the second equality follows by the inductive hypothesis, the third equality follows since  $y(u) = y(v) = y(t) + 1$ , and the last one follows since

$$\text{depth}(G_{st}) = \max\{\text{depth}(G_{su}), \text{depth}(G_{uv}), \text{depth}(G_{vt})\} + 1.$$

*Case 2.*  $f$  is a triangle.

Then  $f = sut$  for some  $u \in V(G_{st})$ . The placement of vertex  $u$  is the same as in Case 1, see Figure 3.3 (right) for an illustration. By a similar computation as in Case 1, it can be verified that  $x(t) - x(u) = |G_{ut} - 1|$ . Further, after ignoring the height of  $\Gamma_{uv}$  and treating  $\Gamma_{vt}$  as  $\Gamma_{ut}$ , the calculation of the height of  $\Gamma_{st}$  is identical to the one presented in Case 1.  $\square$

Note that the proof of Theorem 11 is algorithmic and constructs the final graph drawing. The following corollary is immediate:

**Corollary 2** *Let  $G$  be a maximal outer-1-plane graph. Then  $G$  admits an outer-1-plane straight-line drawing  $\Gamma$  of  $G$  with area  $(\text{depth}(G) + 1) \cdot |G|$ .*

*Proof.* Let  $(s, t)$  be an edge on the outer-face of  $G'$  so that  $\text{depth}(G'_{st}) = \text{depth}(G')$ . Place  $s$  and  $t$  at points  $(0, 0)$  and  $(|G'| - 1, 0)$ , respectively. Now, since  $x(t) - x(s) = |G'| - 1$ , by Theorem 11, we know there exists an outer-1-plane, straight-line drawing  $\Gamma$  of  $G'_{st}$  with width  $|G'|$  and height  $\text{depth}(G'_{st}) + 1$ . By the choice of edge  $(s, t)$ , the area bound follows.  $\square$

We briefly discuss run-time considerations. Note that the edge  $(s, t)$  that should be used to minimize  $\text{depth}(G)$  can naively be found in quadratic time by calculating the depth of  $T_{ab}$  for every edge  $(a, b)$  on the outer-face. It would be interesting to see if this can be achieved in linear time.

Using a similar technique, Garg and Rusu [24] achieved straight-line drawings of outer-1-planar graphs in  $O(d \cdot n^{1.48})$  area. A promising research direction would be to check if their algorithm can be modified to work for outer-1-planar graphs as well.

### 3.4 Drawing Complete Outer-1-Plane Graphs

The above algorithm gives  $O(n \log n)$  area drawings for o1p graphs with depth  $O(\log n)$ , which includes complete o1p graphs. We attempt to improve the area for the subclass of *complete* o1p graphs. Unfortunately, we did not match the  $O(n)$  area bound which is achieved for complete outer-planar graphs by Di Battista and Frati [15]. We instead present an algorithm that achieves drawings in an  $O(n^{0.63}) \times O(n^{0.63})$ -grid, which gives better *aspect ratio* for complete outer-1-plane graphs than the above presented method that achieves an  $O(n) \times O(\log n)$ -grid. We begin the section by reviewing the techniques used by Di Battista and Frati.

### 3.4.1 Review of Di Battista/Frati [15]

Recall that the weak dual graph of a maximal outer-plane graph forms a binary tree. The main drawing technique used by Di Battista and Frati for producing a small area drawing of a maximal outer-plane graph is to first produce a straight-line drawing of its dual tree and then “recover” the drawing of the original graph. To achieve that, the authors defined *star-shaped* drawings of binary trees, and showed that if the dual tree admits a star-shaped drawing in  $f(n)$  area, then the corresponding outer-plane graph can “usually” be drawn in  $O(f(n))$  area as well. Using this technique, they proved that any outer-planar graph admits a planar straight-line embedding-preserving drawing in  $O(n^{1.48})$  area. Further, the area requirements can be improved to  $O(n)$  for the class of complete outer-plane graphs, i.e., outer-plane graphs whose weak dual is a complete binary tree. This result is the main motivation for this chapter.

Let  $T_\rho$  be a binary tree rooted at  $\rho$ . The *leftmost path* and *rightmost path* of  $T_\rho$  are the root-to-leaf paths obtained by repeatedly following the left (right) children, and these paths are denoted by  $L(T_\rho)$  and  $R(T_\rho)$  respectively. More precisely, if  $P := v_1, v_2, \dots, v_k$  is a path such that  $v_1 = \rho$ ,  $v_k$  is a leaf and for all  $i \in \{2, \dots, k\}$ ,  $v_i = v_{i-1}.\text{right}$ , i.e.,  $v_i$  is the right child of  $v_{i-1}$ , we say that  $P$  is the rightmost path of  $T_\rho$ . When there is no ambiguity, we omit the subscript  $\rho$ . Also, once the root is fixed,  $R(v)$  and  $L(v)$  stand for rightmost and leftmost paths of the sub-tree of  $T$  rooted at the vertex  $v$ .

Now, for any  $v \in V(T)$ , the *left-right path* of  $v$  is the path  $v_1, v_2, \dots, v_k$  such that  $v_1 = v$ ,  $v_2 = v_1.\text{left}$  and  $v_2, v_3, \dots, v_k$  is the rightmost path of the sub-tree rooted at  $v_2$ . In other words, it is the path formed by first going to the left child of  $v$  and then following the right children repeatedly until a leaf. Note that when  $v$  has no left child, the left-right path degenerates to a single vertex and when the left-child of  $v$  has no right-child, it forms a single edge. The *right-left path* of  $v$  is symmetrically obtained by first going to the right child of  $v$  and then following the left children until a leaf.

The *left-cycle of neighbors of  $v$*  is the cycle obtained by joining its left-right path  $v_1, v_2, \dots, v_k$  with edge  $(v_k, v_1)$ . If  $k \leq 2$ , then the *left-cycle of neighbors of  $v$*  is a single edge (or a vertex). In a drawing of the tree  $T$ , cycles become polygons (not necessarily simple), so we use  $P_l(v)$  to denote the polygon formed by the left-cycle of  $v$ . The *right cycle of neighbors* and polygon  $P_r(v)$  are analogously defined. See Figure 3.4 (left) for an example of  $P_l(v)$  and  $P_r(v)$ .

The authors now establish a stronger connection between maximal outer-plane graphs and their dual binary trees. Let  $G_{st}$  be a maximal outer-plane graph along with some edge  $(s, t)$  on its outer-face and let  $T$  be its dual binary tree rooted at the face containing the

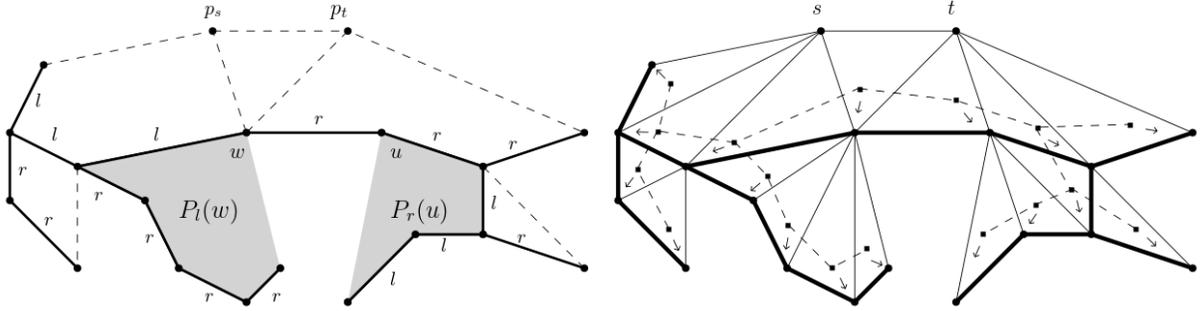


Figure 3.4: Left: An example of a star-shaped drawing of a binary tree rooted at  $w$ . Observe that (SS1) and (SS2) hold for polygons  $P_l(w)$  and  $P_r(u)$  and that  $p_s$  and  $p_t$  satisfy condition (SS3). Right: Illustration of the bijection  $\gamma$ , observe that if  $(f_1, f_2)$  is an edge of the dual tree  $T$ , then  $(\gamma(f_1), \gamma(f_2))$  is an edge of  $G$ .

edge  $(s, t)$ . Also, by removing all edges incident to poles  $s$  and  $t$ , we obtain the *internal graph* of  $G_{st}$ ; we denote it by  $I_{st}$ . It turns out that by adding edges to  $T$ , we can reconstruct graph  $I_{st}$ . Namely, given the dual tree  $T$ , the authors define graph  $T^+$  in the following way: for each vertex  $v \in V(T)$ , connect  $v$  to every vertex  $u$  on the left-right and right-left paths of  $v$  (unless  $u$  is already a neighbor of  $v$ ). They prove that  $I_{st} = T^+$ . The proof technique for this will be important later, so we explain it here.

The key ingredient in the proof that  $I_{st} = T^+$  is a bijection  $\gamma$ <sup>1</sup> between the nodes of  $T$  and the vertices of  $I_{st}$ . To state the  $\gamma$  function explicitly, for every interior face  $f$  (triangle) of  $G_{st}$  we define the *central vertex* of  $f$ . First, if  $f_{st} = stx$  is the face containing the edge  $(s, t)$ , then vertex  $x$  is called the *central vertex* of face  $f_{st}$ . Now, let  $f = abc$  be any non-root face with  $abc$  in clockwise order and w.l.o.g. suppose that  $(a, b)$  is the edge shared with the *parent face* of  $f$ , i.e., the face  $f_p$  that corresponds to the parent of  $f$  in the rooted tree  $T$ . Then vertex  $c$  is called the *central vertex* of  $f$ . Finally, for every interior face  $f$  of  $G_{st}$ , set  $\gamma(f)$  to be the central vertex of  $f$ . See Figure 3.4 (right) for an example. It clearly follows that  $\gamma$  is a bijection between the nodes of the dual tree and the vertices of the internal graph and that for any edge  $(u, v)$  of  $T$ , we have that  $(\gamma(u), \gamma(v))$  is an edge of  $I_{st}$ .

**Observation 2** Let  $G_{st}$  be a maximal outer-plane graph and let  $f = abc$  be any non-root face such that  $(a, b)$  is an edge common to  $f$  and its parent face  $f_p$ .

<sup>1</sup>Note that  $\gamma$  is not a bijection between the internal graph and the dual tree, only the nodes of  $T$  and the vertices of  $I_{st}$ .

1. Either  $a$  or  $b$  must be the central vertex of  $f_p$ .
2. If  $a, b, c$  are ordered in clockwise order, then  $c$  is the central vertex of  $f$ . Further, let  $f_l$  and  $f_r$  be the triangles (if they exist), sharing an edge  $(a, c)$  and  $(b, c)$  with  $f$ , respectively. The central vertex of  $f_l$  is the left child of  $c$ , and the central vertex of  $f_r$  is the right child of  $c$  in the dual tree.

This observation is evident from Figure 3.4 (right), so we omit the proof. Using the correspondence given by the  $\gamma$  function, the authors give an inductive proof of the following lemma:

**Lemma 2**  $T^+ = I_{st}$ .

We are hence interested in drawings of  $T^+$  that permit the insertion of the edges of  $I_{st} \setminus T^+$ . This motivates *star-shaped* drawings of binary trees:

**Definition 2** Let  $T$  be a rooted binary tree and let  $\Gamma$  be a planar straight-line embedding-preserving drawing of  $T$ . We say that  $\Gamma$  is a star-shaped drawing if the following conditions hold:

- SS1** For each vertex  $v \in T$ ,  $P_l(v) = (v_1, \dots, v_k)$  and  $P_r(v) = (v_1, \dots, v_p)$  are simple polygons. Further, for each  $i \in \{3, \dots, k-1\}$  and  $j \in \{3, \dots, p-1\}$ , line segments  $(v, v_i)$  and  $(v, v_j)$  belong to the interior of  $P_l(v)$  and  $P_r(v)$  respectively, except for their endpoints.
- SS2** For each vertex  $v \in T$ , the polygons  $P_l(v)$  and  $P_r(v)$  do not contain any vertices of  $T$  in its interior or on its boundary, except for the vertices of  $P_l(v)$  and  $P_r(v)$ , respectively.
- SS3** There exist points  $p_s, p_t$ , so that we can draw a straight-line segment  $(p_s, p_t)$  and connect  $p_s$  and  $p_t$  to all vertices of  $L(T)$  and  $R(T)$  respectively without creating any crossings with  $\Gamma$ . Also, the polygon formed by  $p_s$  and  $L(T)$  satisfies the conditions imposed on  $P_r(v)$  in (SS1) and (SS2) with respect to  $p_s$ , and the polygon formed by  $p_t$  and  $R(T)$  satisfies the conditions imposed on  $P_l(v)$  in (SS1) and (SS2) with respect to  $p_t$ .

Observe that the tree in Figure 3.4 (left) is drawn star-shaped, if rooted at  $w$ . Condition (SS1) means that polygons  $P_l(v)$  and  $P_r(v)$  have a point that can see all other points (namely, at  $v$ ); such polygons are called *star-shaped* in the literature, and hence the name of the drawings.

Now, given a star-shaped drawing  $\Gamma_T$  of  $T$ , we can obtain a drawing of graph  $T^+$  ( $= I_{st}$ ) by simply adding the edges  $E(T^+) \setminus E(T)$ . The resulting drawing must be planar because of conditions (SS1) and (SS2) of star-shaped drawings. Observe that the final area of  $T^+$  is equal to the area of  $\Gamma_T$ , as we inserted extra edges only. Finally, condition (SS3) allows us to add back vertices  $s$  and  $t$  and their incident edges without creating any crossings with  $\Gamma_T$ . Therefore, we obtain the following result:

**Theorem 12** (Di Battista and Frati [15]) *Let  $G_{st}$  be an  $n$ -vertex maximal outer-plane graph and suppose that its rooted dual tree  $T$  admits a star-shaped drawing with  $f(n)$  area. Then  $G$  admits an outer-planar straight-line drawing such that the area of the drawing of its internal subgraph is  $f(n)$ .*

Notice that the result itself does not guarantee that  $G$  can be drawn in  $f(n)$  area; only its internal subgraph. This is because the points  $p_s$  and  $p_t$  could significantly increase the bounding box of the drawing as we only assumed their existence in (SS3). If the specific algorithm used to produce a star-shaped drawing of  $T$  draws paths  $L(T)$  and  $R(T)$  paths in such a way as to guarantee the existence of  $p_s$  and  $p_t$  so that the area of  $\Gamma \cup \{p_s, p_t\}$  is  $O(f(n))$ , then the drawing of  $G$  would have  $O(f(n))$  area as well.

Finally, we state the theorem which is the main motivation for this chapter:

**Theorem 13** (Di Battista and Frati [15]) *Every  $n$ -vertex complete outer-plane graph  $G$  admits an outer-planar straight-line grid drawing  $\Gamma$  such that both the height and the width of  $\Gamma$  are  $O(\sqrt{n})$ .*

As discussed above, to prove the theorem it suffices to give a star-shaped drawing of the complete binary tree that has width and height  $O(\sqrt{n})$  even when including the points  $p_s$  and  $p_t$  used for the poles. The technique used to produce such a star-shaped drawing  $\Gamma_h$  of a complete binary tree of height  $h$  is to recursively draw all four “grand-child” subtrees of height  $h - 2$  and obtain drawings  $\Gamma_{h-2}$ . By flipping two such drawings “inside”, they obtain a square-like drawing  $\Gamma_h$  of  $G$  in linear area. The output of their algorithm for a binary tree of height three is shown in Figure 3.5.

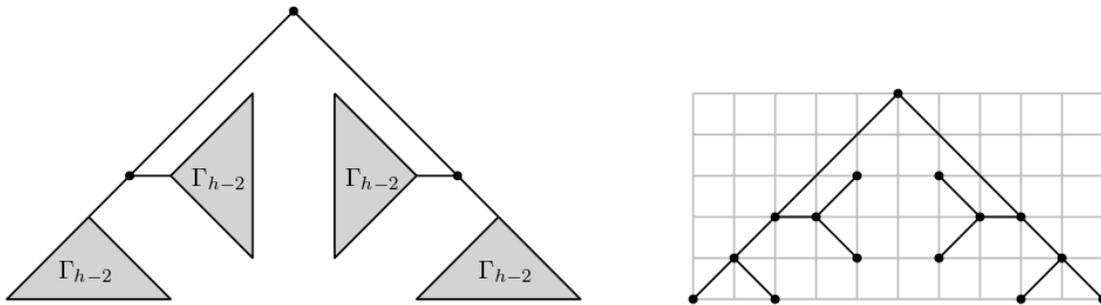


Figure 3.5: Output of Di Battista/Fratti subroutine for drawing a star-shaped binary tree of height three.

### 3.4.2 Generalization to Complete Outer-1-Plane Graphs

To achieve small area drawings of complete o1p graphs, we extend the technique used by Di Battista and Frati. Let  $G_{st}$  be a complete o1p graph and recall that by definition, the dual of  $skel(G_{st})$  is a complete ternary tree  $T$ . Recall that in complete ternary trees every node except for the leaves has three children, so every face of  $skel(G_{st})$  must be a quadrangle except possibly for faces dual to leaves of  $T$ . For simplicity, to avoid handling the leaf triangle faces, we restrict ourselves to *quadrangular* o1p graphs only, i.e., o1p graphs whose skeleton consists of quadrangles only. This assumption does not violate the asymptotic bounds on the area of the drawings as we can always add one new vertex for every leaf face that is a triangle to form a quadrangular graph that contains the original graph (the new graph has  $O(n)$  vertices). Also, analogous to outer-plane graphs, we use  $I_{st}$  to denote the *internal graph* of  $G_{st}$ , the graph obtained by removing vertices  $s$  and  $t$  and their incident edges.

We expand on the approach from [15] in the following way:

*Step 1:* Define zig-zag star-shaped drawings of binary trees.

*Step 2:* Appropriately add an edge to each quadrangle of  $skel(G)$  to produce a specific *half-skeleton*  $M(G)$ , a maximal outer-planar graph.

*Step 3:* Show that if the dual tree  $T$  of  $M(G)$  admits a zig-zag star-shaped drawing in  $f(n)$  area, then the internal graph  $I_{st}$  of  $G$  admits an outer-1-plane straight-line drawing in  $f(n)$  area.

*Step 4:* Give an algorithm that produces small area zig-zag star-shaped drawings of binary trees.

*Step 5:* Finally, draw the outer-1-plane graph.

The algorithm will produce a straight-line drawing in an  $O(2^h) \times O(2^h)$ -grid of any quadrangular graph  $G$ , where  $h$  is the height of the dual ternary tree of  $skel(G)$ . As an immediate corollary we will obtain drawings of width and height  $O(2^{\lceil \log_3(2n) \rceil}) = O(n^{0.63})$  for the subclass of complete quadrangular o1p graphs, since complete ternary trees have height  $\lceil \log_3(2n) \rceil$ .

### Step 1: Zigzag Star-shaped Drawings

We first extend the definition of star-shaped drawings of binary trees by adding several convexity conditions:

**Definition 3** *Let  $T$  be a binary tree. We say that a planar straight-line embedding-preserving drawing  $\Gamma$  of  $T$  is weakly zigzag star-shaped if the following conditions hold:*

**ZZS1**  $\Gamma$  satisfies conditions (SS1) and (SS2) of star-shaped drawings ((SS3) is omitted).

**ZZS2** For every vertex  $v \in V(T)$  with  $\langle v = w_1, w_2, \dots, w_k \rangle$  and  $\langle v = u_1, u_2, \dots, u_t \rangle$  as its left-right and right-left paths, the following conditions hold:

1. For all even  $i, 2 \leq i \leq k - 2$ ,  $vw_iw_{i+1}w_{i+2}$  is a strictly convex polygon.
2. For all even  $i, 2 \leq i \leq t - 2$ ,  $vu_iu_{i+1}u_{i+2}$  is a strictly convex polygon.

Observe that *weakly zigzag star-shaped drawings* are not required to satisfy condition (SS3). They will only be useful during recursion. We now strengthen the original pole recovery condition (SS3) in the following way:

**Definition 4** *Let  $T$  be a binary tree. We say that a planar straight-line embedding-preserving drawing  $\Gamma$  of  $T$  is zigzag star-shaped if it is weakly zigzag star-shaped and if the following condition holds:*

**ZZS3** Let  $L(\Gamma) = v_1, v_2, \dots, v_k$  and  $R(\Gamma) = u_1, u_2, \dots, u_t$  be the leftmost and rightmost paths of  $\Gamma$ . Then there exist points  $p_s$  and  $p_t$  that satisfy (SS3) condition of star-shaped, and

1.  $p_s p_t u_2 u_1$  is a strictly convex polygon.
2. For all odd  $i, 1 \leq i \leq k - 2$ ,  $p_s v_i v_{i+1} v_{i+2}$  is a strictly convex polygon.
3. For all even  $i, 2 \leq i \leq t - 2$ ,  $p_t u_i u_{i+1} u_{i+2}$  is a strictly convex polygon.

See Figure 3.6 for an example of a zigzag star-shaped binary tree drawing.



2. Otherwise, the parent face of  $f$ , say  $f_p$ , will already have been made into a triangle. Let  $(v, w)$  be the edge common to  $f$  and  $f_p$ . By Observation 2 one of  $v, w$  will be the central vertex of  $f_p$ , say this is  $w$ . Add to  $M(G)$  the diagonal that is incident to  $v$  (not incident to the central vertex of  $f_p$ ) to face  $f$ . The other end of this diagonal (say  $y$ ) becomes the central vertex of the newly created face  $f_1$  of  $M(G)$ , and  $y$  becomes a child of  $w$  in the dual tree  $T_M$  of  $M(G)$ . It is the left (right) child, if  $f_1$  consists of  $\{v, w, y\}$  in clockwise (counter-clockwise) order. The fourth vertex of  $f$ , say  $x$ , becomes the central vertex of the other created face  $f_2$ , it is a child of  $y$  and a left/right child if and only if  $y$  is the left/right child.
3. Recurse on the children of  $f$ .

See Figure 3.7 for an example of  $M(G)$  and its dual binary tree  $T$  with every node mapped to the central vertex of its dual face, illustrating the  $\gamma$  bijection. Consider the face  $f = \{v, w, y, x\}$  from Figure 3.7 and observe that the central vertex of its parent face is  $w$ . Hence, the edge  $(v, y)$  is inserted, i.e., the diagonal *not* incident to  $w$ .

### Properties of $M(G)$

Recall that  $I_{st}$  stands for the internal graph of outer-1-plane graph  $G_{st}$ , so we use  $I_M$  to denote the internal graph of  $M(G)$ . Also recall that  $\gamma$  is a bijection that maps the nodes of the dual tree of  $M(G)$  (a maximal outer-plane graph) to the vertices of  $I_M$ . Such that if  $(u, v)$  is an edge of the dual binary tree of  $M(G)$ , then  $(\gamma(u), \gamma(v))$  is an edge of the internal graph  $I_M$ . So, we use  $T_M$  to denote the *dropdown dual* tree formed by vertices  $\gamma(u)$  and edges  $(\gamma(u), \gamma(v))$ . This way we simultaneously treat any vertex  $v$  as a vertex of the internal graph and as a node of the dual tree of  $M(G)$ . The following two lemmas list the properties of  $M(G)$  that we need:

**Lemma 3** *Let  $G_{st}$  be a quadrangular o1p graph and let  $T_M$  be the dropdown dual tree of  $M(G)$ . For every interior face  $f$  of  $skel(I_{st})$ , there exists a vertex  $v \in V(T_M)$  so that the left-right or right-left path of  $v$ , say  $v_1, v_2, \dots, v_k$ , contains all vertices of  $f$ , and  $f = \{vv_i v_{i+1} v_{i+2}\}$  for some even  $i \in \{1, \dots, k\}$ .*

*Proof.* We prove a slightly stronger claim. In addition to the lemma statement, we show that the number of vertices on any left-right/right-left path in  $T_M$  is even (except for the single vertex paths).

We proceed by induction on the number of faces  $F$  of  $skel(G_{st})$ .

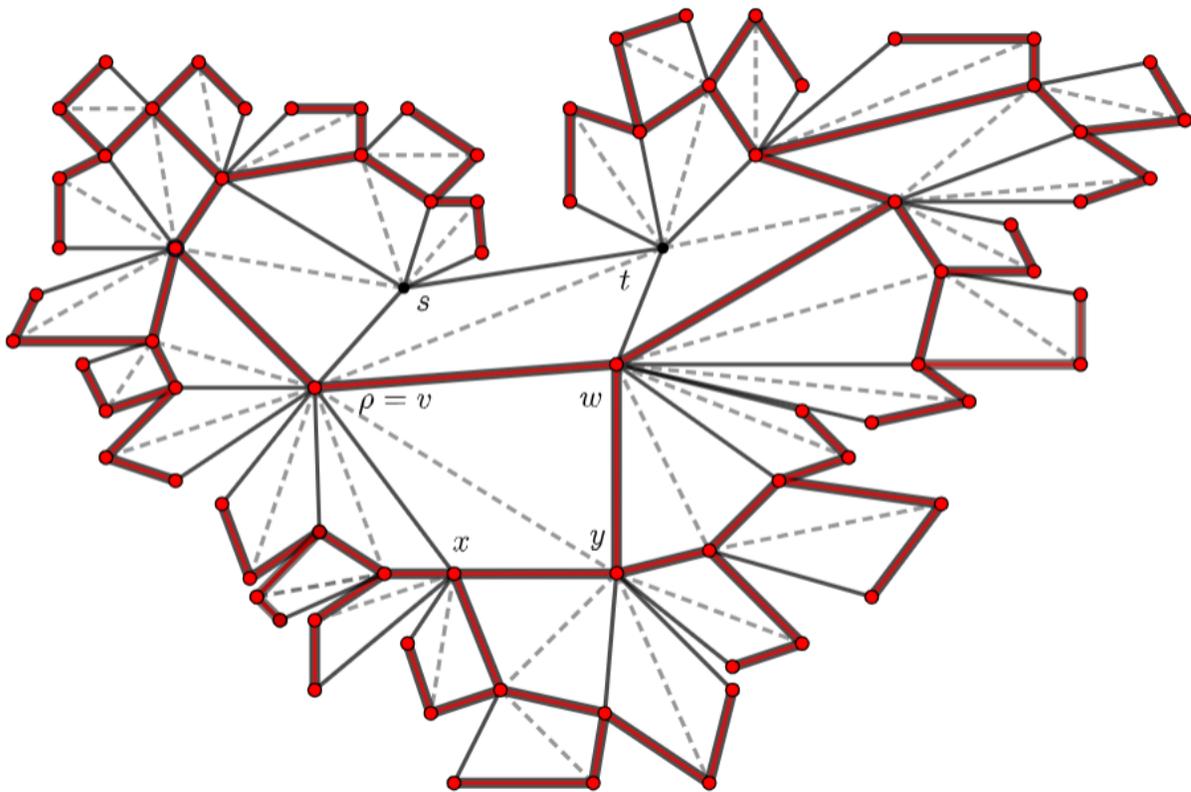


Figure 3.7: An example of  $M(G)$  on a complete o1p graph  $G_{st}$ . The dual tree of  $M(G)$  rooted at  $\rho$  is shown with thick red lines.

*Base Case.* If  $F = 1$ , then  $skel(G_{st}) = \{s, t, x, \rho\}$  for some vertices  $x$  and  $\rho$  since  $G_{st}$  is quadrangular. Hence  $I_{st}$  has no interior faces so the result for faces holds vacuously. We inserted edge  $(t, \rho)$ , which makes  $\rho$  the root of  $T_M$  and  $x$  its right child. So, the only non-trivial right-left path is  $\rho x$ , and has size 2.

*Inductive Step.* Let  $G_{st}$  be a quadrangular o1p graph with  $F > 1$  faces and suppose inductively that the (stronger) claim holds for the skeleton of any quadrangular o1p graph with fewer than  $F$  faces. See Figure 3.8 for a demonstration of the proof below.

Let  $f$  be a face of  $skel(I_{st})$  with vertices enumerated in clockwise order. If  $f$  is not a leaf face, then let  $G'$  be the the subgraph of  $skel(G_{st})$  obtained by removing one leaf face of  $skel(G_{st})$ . Since  $f$  is not a leaf, it is also a face of  $G'$ . Since  $G'$  has  $F - 1$  faces, induction applies, and so the claim holds for  $f$ . We now assume  $f$  is a leaf face. Without loss of generality suppose that edge  $(a, b)$  of  $f$  is shared with the face  $f_p$  that is the parent face of  $f$ . By Observation 2, we know that the central vertex of  $f_p$  is one of  $\{a, b\}$ .

*Case 1.* The central vertex of  $f_p$  is  $b$ . See Figure 3.8.

By construction of  $M(G)$ , the diagonal not incident to  $b$  was inserted in  $f$ , i.e., edge  $(a, c)$ . So, let  $f_1 = abc$  and  $f_2 = acd$  be the two new triangle faces. Since  $a, b, c$  are ordered clockwise and since  $(a, b)$  is shared with  $f_p$ , we know by Observation 2 (2) that the left child  $\gamma(f_p).left$  of  $\gamma(f_p)$  is  $\gamma(f_1)$  and likewise  $\gamma(f_1).left = \gamma(f_2)$ . So,  $\gamma(f_p), \gamma(f_1), \gamma(f_2)$  are on the same right-left path. This is not the leftmost path as we assumed that  $f$  is a face of  $skel(I_{st})$ , while the faces incident to the leftmost/rightmost path contain  $s$  or  $t$ .

Let  $P := u_1, u_2, \dots, u_t$  be the right-left path containing  $\gamma(f_p), \gamma(f_2), \gamma(f_1)$  and note that  $\gamma(f_2) = u_t, \gamma(f_1) = u_{t-1}$  and  $\gamma(f_p) = u_{t-2}$ , since  $f$  is a leaf. Note that by Lemma 2 (Di Battista and Frati), we know that  $I_M = T^+$ . Recall that  $T^+$  is constructed by connecting every vertex  $v$  to all the vertices on its left-right and right-left paths. So, if  $(x, y)$  is an edge on the right-left path of  $v$  and  $x.left = y$ , then  $\{v, x, y\}$  is a face of  $M(G)$ .

Now, since  $(\gamma(f_p), \gamma(f_1))$  is on the right-left path of  $u_1$  and since  $\gamma(f_p).left = \gamma(f_1)$ , we know that  $\{u_1, \gamma(f_p), \gamma(f_1)\}$  is a face, hence  $a = u_1$ . So, since  $\gamma(f_p), \gamma(f_1), \gamma(f_2)$  are on the right-left path of  $u_1$  we have that  $f = \{u_1, \gamma(f_p), \gamma(f_1), \gamma(f_2)\} = \{u_1, u_{t-2}, u_{t-1}, u_t\}$ . By the strengthened inductive hypothesis we know that  $P_{sub} := u_1, u_2, \dots, u_{t-2}$  is of even size, say  $i := t - 2$ , hence  $f = \{u_1, u_i, u_{i+1}, u_{i+2}\}$ .

Lastly, we show that the stronger claim holds. Single vertex paths are excluded from the stronger claim and the statement holds for left-right and right-left paths of size two. Now, recall that we did induction on the number of faces and observe that  $P$  is the only new left-right or right-left path of size two or greater created by adding face  $f$ . Since two vertices are added to  $P_{sub}$ , path  $P$  must be of even length.

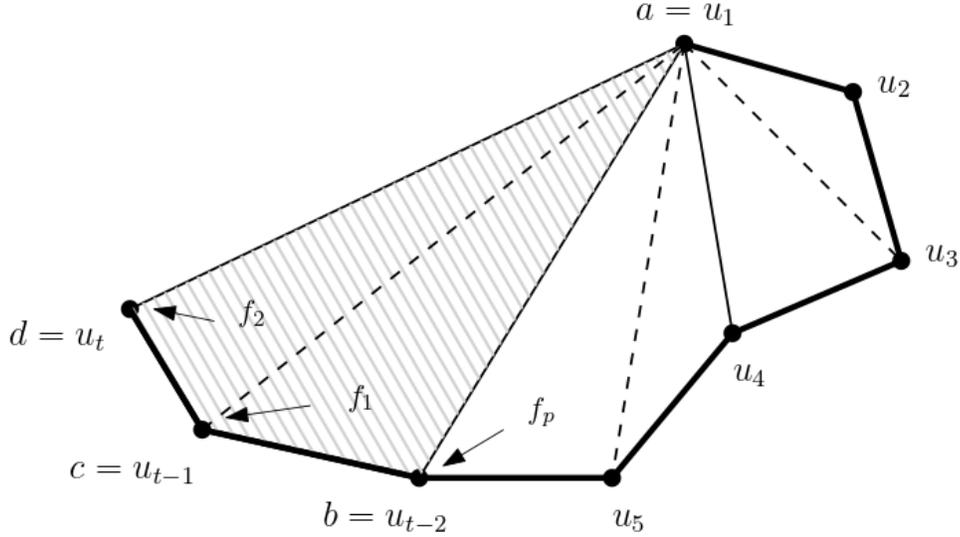


Figure 3.8: Proof of Lemma 3, face  $f$  is the dashed region.

*Case 2.* The central vertex of  $f_p$  is  $a$ .

This case is symmetric to Case 1. This time, edge  $(b, d)$  is the inserted diagonal in  $M(G)$  and path  $P$  will be a left-right path, as opposed to the right-left path in Case 1.  $\square$

It remains to state a lemma for the faces incident to the poles  $s$  and  $t$ , faces not belonging to the internal graph:

**Lemma 4** *Let  $G_{st}$  be a quadrangular  $o1p$  graph and let  $T_M$  be the dropdown dual tree of  $M(G)$ . Further, let  $L(T_M) := v_1, v_2, \dots, v_k$  and  $R(T_M) = u_1, u_2, \dots, u_t$  be the leftmost and rightmost paths of  $T_M$ , respectively. Then for every face  $f$  of  $\text{skel}(G)$  containing  $s$  or  $t$  vertices, we have that either*

1.  $f = \{s, t, u_2, u_1\}$ , or
2.  $f = \{s, v_i, v_{i+1}, v_{i+2}\}$  for some odd  $i \in \{1, \dots, k\}$ , or
3.  $f = \{t, u_i, u_{i+1}, u_{i+2}\}$  for some even  $i \in \{2, \dots, t\}$ .

*Proof.* Observe that  $\{s, t, u_2, u_1\}$  is a face ( $\{s, t, w, \rho\}$  in Figure 3.7), which is why the indexing is shifted for the faces incident to  $s$ . The essence of the argument is similar to

Lemma 3, except we use leftmost path and rightmost path as opposed to left-right/right-left paths. The formal proof is by induction and similar to the previous lemma, hence omitted here.  $\square$

### Step 3: Recovering a Drawing of $G_{st}$

We now state a result analogous to Theorem 12:

**Theorem 14** *Let  $G_{st}$  be a quadrangular o1p graph and let  $T_M$  be the dual binary tree of  $M(G)$ . If  $T_M$  admits a zigzag star-shaped drawing in  $f(n)$  area, then  $G$  admits an outer-1-planar straight-line drawing such that the internal graph  $I_{st}$  is drawn in  $f(n)$  area.*

*Proof.* Let  $\Gamma_{T_M}$  be a zigzag star-shaped drawing of  $T_M$ . Also, let  $\Gamma$  be the drawing obtained by adding to  $\Gamma_{T_M}$  the two points  $p_s$  and  $p_t$  that we know exist from condition (ZZS3). Place  $s$  at  $p_s$  and  $t$  at  $p_t$ . The two points also satisfy (SS3), so by Theorem 12  $\Gamma$  is an embedding-preserving planar straight-line drawing of  $M(G)$ , and  $I_{st}$  is drawn in  $f(n)$  area. It remains to show that all faces of  $skel(G)$  are drawn strictly convex, so let  $f$  be an arbitrary face of  $skel(G)$ . If  $f$  belongs to  $skel(I_{st})$ , then it must be strictly convex by Lemma 3 and condition (ZZS2). In the other case, when  $f$  is incident to  $s$  or  $t$ , by Lemma 4 and condition (ZZS3),  $f$  must also be strictly convex.  $\square$

Similar to the result by Di Battista and Frati (Theorem 12), the drawing of the binary tree is sufficient for recovering the internal graph in the same area. To achieve  $O(f(n))$  area for the entire graph  $G$ , we need to ensure that the axis aligned rectangle containing both the binary tree drawing and the poles  $s$  and  $t$  has  $O(f(n))$  area, hence:

**Corollary 3** *Let  $G_{st}$  be a quadrangular o1p graph and let  $T_M$  be the dual binary tree of  $M(G)$ . If  $T_M$  admits a zig-zag star-shaped drawing  $\Gamma$  such that the bounding box containing  $\Gamma \cup \{p_s, p_t\}$  has  $f(n)$  area (where  $p_s$  and  $p_t$  are the points given by (ZZS3)), then  $G$  admits a straight-line outer-1-planar drawing in  $f(n)$  area.*

### Step 4: Drawing Zig-zag Star-shaped Binary Trees

Next, our aim is to show that any binary tree  $T$  of height  $h$  admits a zig-zag star-shaped drawing so that both the height and the width of the drawing are  $O(2^{h/2})$ . Then, later in Step 5, we will apply Corollary 3 to recover the original outer-1-plane graph. We begin by defining different types of drawings used by our algorithm and state some of their properties.

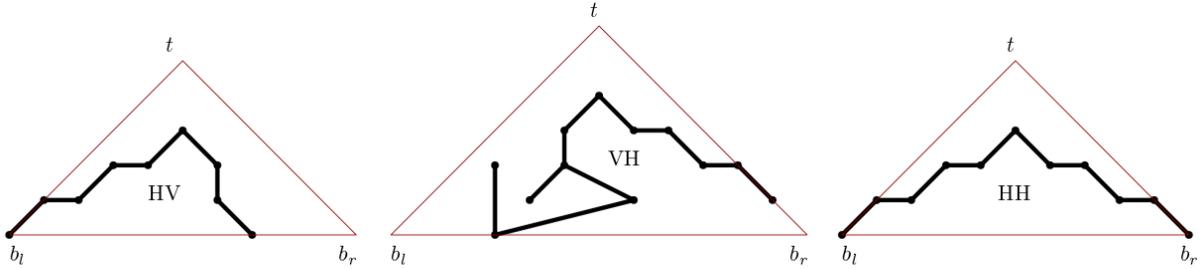


Figure 3.9: From left to right: HV, VH and HH drawings and their enclosing triangles.

### Drawing Types and Their Properties

Let  $\Gamma_P$  be a planar straight-line drawing of a path  $P = v_1, \dots, v_n$ . We say that  $\Gamma_P$  is a *diagonal-horizontal left* drawing of  $P$  if for all  $i \in \{1, \dots, n-1\}$ , if  $i$  is odd, edge  $v_i v_{i+1}$  is drawn downward with slope  $\pi/4$ , otherwise  $v_i v_{i+1}$  is drawn as a unit-length horizontal segment. So, the first edge is drawn downward diagonally to the left, and all subsequent ones alternate between horizontal and diagonal. Analogously, the first edge of a *diagonal vertical left* path is drawn downward with slope  $\pi/4$ , the second edge as a vertical line segment etc., however we do not require vertical line segments to be of unit-length. *Diagonal vertical right* and *diagonal horizontal right* drawings are defined symmetrically. Recall that  $L(T)$  and  $R(T)$  stand for the leftmost and rightmost paths of  $T$ . We distinguish between three types of binary tree drawings:

1. An *HV*<sup>2</sup> drawing is one where  $L(T)$  is drawn diagonal-horizontal left and  $R(T)$  is drawn diagonal-vertical right.
2. A *VH* drawing is one where  $L(T)$  is drawn diagonal-vertical left and  $R(T)$  is drawn diagonal-horizontal right.
3. An *HH* drawing is one where  $L(T)$  is drawn diagonal-horizontal left and  $R(T)$  is drawn diagonal-horizontal right.

See Figure 3.9 for an example of each type of drawing. We refer to any of the above drawings as *staircase drawings*. Let  $\Gamma_\rho$  be a staircase drawing of a binary tree rooted at  $\rho$ . We use  $Enc(\Gamma)$  to denote the smallest right isosceles triangle containing  $\Gamma_\rho$  so that the right angle point of  $Enc(\Gamma)$  and the root  $\rho$  have equal  $x$  coordinates. Further, we use

<sup>2</sup>Not related to a standard “hv-drawing”, a common type of drawing in the binary tree graph drawing literature. For instance see [10] for hv-drawings of Fibonacci trees.

$t(Enc(\Gamma)), b_l(Enc(\Gamma)), b_r(Enc(\Gamma))$  to denote the top point, and left and right points of the base edge of  $Enc(\Gamma)$ , respectively. We usually say “place the enclosing triangle at a certain position”; by this we mean to move  $\Gamma$  so that  $Enc(\Gamma)$  is in the given position. For example, when saying “rotate  $Enc(\Gamma)$  by  $90^\circ$  and place  $t(Enc(\Gamma))$  at coordinate  $(0, 0)$ ”, we assume that  $\Gamma$  is rotated and translated along with  $Enc(\Gamma)$ .

Now, let  $\Gamma$  be a staircase drawing and let  $v_L$  and  $v_R$  be the leaves of  $L(\Gamma)$  and  $R(\Gamma)$ , respectively. Let  $d_L$  be a downward (infinite) diagonal with slope  $\pi/4$  starting at vertex  $v_L$  and let  $d_R$  be a downward (infinite) diagonal with slope  $-\pi/4$  starting at vertex  $v_R$ . We call the unbounded region below or on the poly-line formed by concatenating  $d_L, L(\Gamma), R(\Gamma)$  and  $d_R$ , the *trapped region* of  $\Gamma$ .

**Definition 5** *We say that a staircase drawing  $\Gamma$  of a binary tree  $T$  is extreme if every vertex of  $T$  is drawn in the trapped region of  $\Gamma$ .*

See Figure 3.10 (left) for an example of a non-extreme drawing. Extreme staircase drawings are much more restrictive than the general staircase drawings and will have several desirable properties. For example, we now make an observation about the distance between the root of the tree and the top of the enclosing triangle in extreme staircase drawings, which will be needed for analysing the area of the drawing.

**Lemma 5** *Let  $\Gamma$  be an extreme HV drawing or an extreme VH drawing of a binary tree  $T$  and let  $k$  be the number of edges on the diagonal-horizontal path. Then  $|y(t(Enc(\Gamma))) - y(\rho)| = \lfloor \frac{k}{2} \rfloor$ . If both  $L(T)$  and  $R(T)$  are diagonal-horizontal paths with  $k_L$  and  $k_R$  edges, then  $|y(t(Enc(\Gamma))) - y(\rho)| = \max\{\lfloor k_L/2 \rfloor, \lfloor k_R/2 \rfloor\}$ .*

*Proof.* We only show the claim for an extreme HV drawing  $\Gamma$ , the other arguments are similar. Since  $\Gamma$  is extreme, all vertices of  $\Gamma$  are in the trapped region of  $\Gamma$  and the leaf of the leftmost (diagonal-horizontal) path must touch the left side of  $Enc(\Gamma)$ , i.e., line segment  $(t(Enc(\Gamma)), b_l(Enc(\Gamma)))$ . Therefore  $|y(t(Enc(\Gamma))) - y(\rho)|$  is determined by the total length of the horizontal line segments in  $L(\Gamma)$ . Since horizontal line segments have unit length by assumption, and half of the edges of  $L(\Gamma)$  are drawn horizontally, the claim holds. See Figure 3.10 (right) for an example.  $\square$

For any straight line drawing  $\Gamma$ , we say that a vertex  $v$  *sees* vertex  $u$ , if drawing edge  $(u, v)$  would intersect any edges or vertices of  $\Gamma$ . The primary motivation for defining enclosing triangles are their visibility properties which will be needed for recursively merging staircase drawings in our drawing algorithm. Here we state four visibility properties of the enclosing triangles (two up to symmetry) that all extreme staircase drawings have:

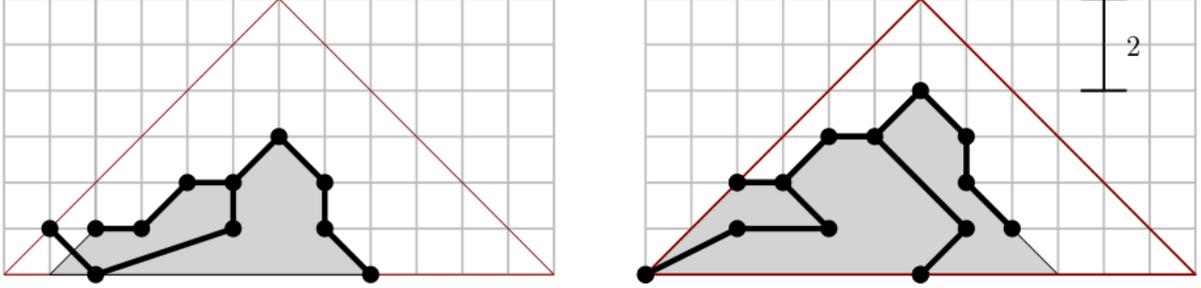


Figure 3.10: Left: An example of non-extreme HV drawing. Right: Illustration that for extreme HV drawings  $|y(t(\text{Enc}(\Gamma)) - y(\rho)|$  is the number of horizontal edges on the leftmost path.

**Lemma 6** (*Enclosing Triangle Visibility Lemma*) *Let  $T$  be a binary tree rooted at  $\rho$ , let  $\Gamma$  be an extreme staircase drawing of  $T$  and let  $t = t(\text{Enc}(\Gamma))$ ,  $b_l = b_l(\text{Enc}(\Gamma))$ , and  $b_r = b_r(\text{Enc}(\Gamma))$ . Then,*

- V1:** *If  $R(T)$  is drawn diagonal-vertical and if  $p$  is any point in the south-east (SE) region of  $\text{Enc}(\Gamma)$ , i.e., point  $p$  is on the right side (strictly) of the line containing points  $t$  and  $b_r$  and  $y(p) \leq y(b_r)$ , then  $p$  sees all vertices of  $R(T)$ .*
- V2:** *If  $L(T)$  is drawn diagonal-vertical and if  $p$  is any point in the south-west (SW) region of  $\text{Enc}(\Gamma)$ , i.e., point  $p$  is any point on the left side (strictly) of the line containing points  $t$  and  $b_l$  and  $y(p) \leq y(b_l)$ , then  $p$  sees all vertices of  $L(T)$ .*
- V3:** *If  $R(T)$  is drawn diagonal-horizontal and if  $p$  is any point in the north-east (NE) region of  $\text{Enc}(\Gamma)$ , i.e., point  $p$  is any point so that  $x(p) \geq x(t)$  and  $y(p) \geq y(t)$  (and  $p \neq t$ ), then  $p$  sees every vertex on  $R(T)$ .*
- V4:** *If  $L(T)$  is drawn diagonal-horizontal and if  $p$  is any point in the north-west (NW) region of  $\text{Enc}(\Gamma)$ , i.e., point  $p$  is any point so that  $x(p) \leq x(t)$  and  $y(p) \geq y(t)$  (and  $p \neq t$ ), then  $p$  sees every vertex on  $L(T)$ .*

*Proof.* We will only prove (V4) here, the other arguments are similar. Let  $p$  be a point in the NW region, and let  $v$  be an arbitrary vertex on  $L(T)$ . Observe that the NW region is outside the trapped region and  $v$  is on the boundary of the trapped region. Since  $x(p) \leq x(t)$ , segment  $\overline{pv}$  could hence intersect  $\Gamma$  only if it intersects an edge of  $L(T)$  or the left-downward diagonal from the leftmost leaf. But all lines from  $v$  that intersect edges

of  $L(T)$  have slope between zero and one and do not intersect the NW region. So line segment  $\overline{pv}$  does not intersect  $\Gamma$ .  $\square$

See Figure 3.11 (left) for an example and observe that an HV drawing shown in the figure is extreme. Figure 3.11 (right) demonstrates that the Lemma 6 may not hold for non-extreme staircase drawings.

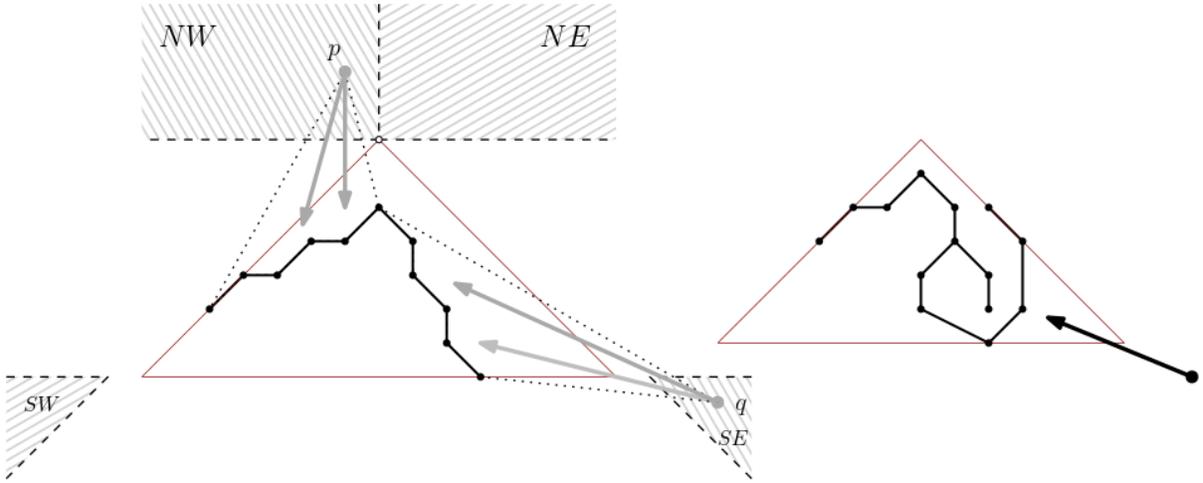


Figure 3.11: Left: Illustration of the four regions from Lemma 6 for an extreme  $HV$  drawing. Point  $q$  is an example of property (V1) and point  $p$  is an example of property (V4). Right: An example of a non-extreme  $HV$  drawing that does not satisfy property (V4).

### The Lifting Operation

Let  $\Gamma_\rho$  be a staircase drawing of  $T$ . Recall that the points  $\rho$  and  $t(Enc(\Gamma))$  have equal  $x$  coordinates. However, to recursively merge the sub-tree drawings, we may need these points to coincide. We therefore define the *left lift* of  $\Gamma$  to be the drawing obtained by lifting the root  $\rho$  along with the sub-tree of  $T$  rooted at the left child of  $\rho$  ( $T_{\rho.\text{left}}$ ), to the top of the enclosing triangle. More precisely,

- Let  $d := y(t(Enc(\Gamma))) - y(\rho)$ .
- For each  $v \in T_{\rho.\text{left}} \cup \rho$  in  $\Gamma$ , set  $y^{\text{new}}(v) := y(v) + d$ .

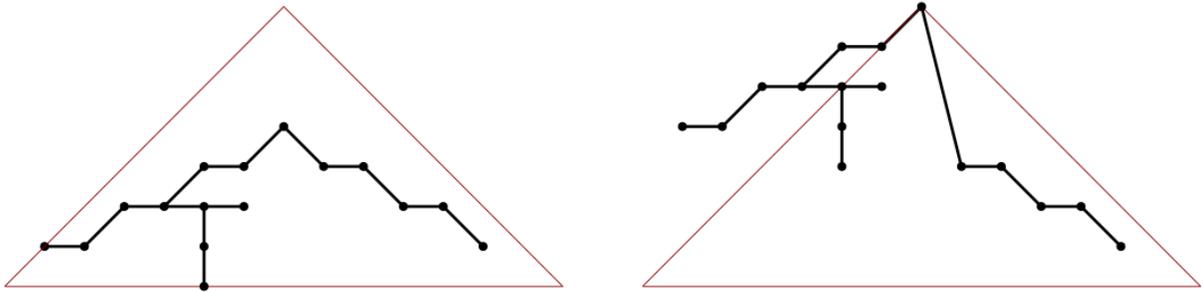


Figure 3.12: An example of the left lift operation.

We denote the resulting drawing by  $\Gamma^L$  (not to be confused with  $L(\Gamma)$ ). The right lift is defined symmetrically. See Figure 3.12 for an example. We now state the *Left Lift Invariant*, the list of important properties of staircase drawings we want to be preserved by  $\Gamma^L$ . The properties (L2) to (L6) are easily shown to hold for the left lift of any extreme HH drawing, while property (L1) does not always hold, but will hold for the specific drawings produced by our drawing algorithm, as will be shown in Theorem 15. Also, recall that weakly zigzag star-shaped are zigzag star-shaped drawings that do not have to satisfy conditions (SS3) or (ZZS3) (pole recovery conditions).

**Definition 6** *Let  $\Gamma$  be a weakly zig-zag star-shaped HH drawing of  $T$  rooted at  $\rho$  and let  $t = t(\text{Enc}(\Gamma))$  and  $b_r = b_r(\text{Enc}(\Gamma))$ . We say  $\Gamma$  satisfies the left lift invariant if the following conditions hold:*

- L1**  $\Gamma^L$  is a weakly zig-zag star-shaped drawing.
- L2** The leftmost path  $L(\Gamma^L)$  is diagonal-horizontal.
- L3** If  $p$  is any point in the NE region of  $\text{Enc}(\Gamma)$ , then  $p$  sees every vertex on  $R(\Gamma^L)$ , the rightmost path of the lifted drawing.
- L4**  $R(\Gamma^L)$  is contained within  $\text{Enc}(\Gamma)$ .
- L5** If  $R(\Gamma^L) = \langle v_1, v_2, \dots, v_k \rangle$ , then  $v_i$  sees  $v_{i+2}$  for all  $1 \leq i \leq k - 2$ .
- L6** All vertices of  $\Gamma^L$  are to the right of the poly-line formed by  $L(\Gamma^L)$  and the downward diagonal with slope  $\pi/2$  starting at the leaf of  $L(T)$ .

Note that  $\Gamma^L$  is not an  $HH$  drawing since the edge  $(\rho, \rho.\text{right})$  is not at  $45^\circ$  anymore. So, in (L6), we cannot claim that  $\Gamma^L$  is an extreme  $HH$  drawing, however for our purposes (L6) is sufficient. Also, observe that in visibility property (L3), the region of point  $p$  is defined with respect to the enclosing triangle of  $\Gamma$ , not  $\Gamma^L$ .

## The Drawing Algorithm

We next present the tree drawing algorithm. Recall that the height of any graph drawing  $\Gamma$  is defined to be the number of rows occupied by  $\Gamma$ .

**Theorem 15** *Let  $T$  be a binary tree with height  $h$ . Then:*

**Invariant 1:**  *$T$  has a weakly zigzag star-shaped extreme  $HV$  drawing  $\Gamma$  such that  $\text{height}(\text{Enc}(\Gamma)) \leq 10 \cdot 2^{\lfloor h/2 \rfloor} - h - 7$ .*

**Invariant 2:**  *$T$  has a weakly zigzag star-shaped extreme  $VH$  drawing  $\Gamma$  such that  $\text{height}(\text{Enc}(\Gamma)) \leq 10 \cdot 2^{\lfloor h/2 \rfloor} - h - 7$ .*

**Invariant 3:**  *$T$  has a weakly zigzag star-shaped extreme  $HH$  drawing  $\Gamma$  such that  $\text{height}(\text{Enc}(\Gamma)) \leq 10 \cdot 2^{\lfloor h/2 \rfloor} - h - 7$  and  $\Gamma$  satisfies both the left and the right lift invariant.*

*Proof.* First, augment  $T$  to a complete binary tree of height  $h$ .

*Base case.* If  $h = 0$ , then place  $T$ , a single vertex, at  $(0, 0)$ . The enclosing triangle of the single vertex is the vertex itself, hence its height is  $1 < 10 \cdot 2^{\lfloor 0/2 \rfloor} - 0 - 7 = 3$ . If  $h = 1$ , place the root of  $T$  at  $(1, 1)$ , its left child at  $(0, 0)$  and the right child at  $(2, 0)$ . The height of the resulting drawing  $\Gamma$  is 2 and  $10 \cdot 2^{\lfloor 1/2 \rfloor} - 1 - 7 = 2 \geq 2$ . All the invariant properties are easily verified.

*Induction Step.* If  $h \geq 2$ , assume inductively that the theorem holds for any binary tree of height at most  $h - 1$ . Let  $\rho$  be the root of  $T$  and denote the grandchildren of  $\rho$  by  $v_{ll}, v_{lr}, v_{rl}, v_{rr}$  where  $v_{ll} := \rho.\text{left}.\text{left}$ ,  $v_{lr} := \rho.\text{left}.\text{right}$ ,  $v_{rl} := \rho.\text{right}.\text{left}$  and  $v_{rr} := \rho.\text{right}.\text{right}$ .

*Construction 1.*

We draw  $T$  satisfying Invariant 1. Note that since  $h \geq 2$ , the heights of each of  $T_{v_{ll}}, T_{v_{lr}}, T_{v_{rl}}$  and  $T_{v_{rr}}$  are exactly  $h - 2 \geq 0$ . By the inductive hypotheses we can do the following:

- Draw  $T_{v_{ll}}$  using Invariant 3 to get  $\Gamma_{v_{ll}}$ .
- Draw  $T_{v_{lr}}$  using Invariant 1 to get  $\Gamma_{v_{lr}}$ .
- Draw  $T_{v_{rl}}$  using Invariant 2 to get  $\Gamma_{v_{rl}}$ .
- Draw  $T_{v_{rr}}$  using Invariant 1 to get  $\Gamma_{v_{rr}}$ .

Let  $h_{max} = \max\{\text{height of } Enc(\Gamma_w) | w \in \{v_{ll}, v_{lr}, v_{rl}, v_{rr}\}\}$ . Now, for each  $w \in \{v_{ll}, v_{lr}, v_{rl}, v_{rr}\}$ , let  $E_w$  be the right isosceles triangle that encloses  $\Gamma_w$ , has  $t(Enc(\Gamma_w))$  at its top and has height  $h_{max}$ . Put differently, increase the height of  $\Gamma_w$  by adding empty rows below it. We abbreviate  $E_{v_{ll}}, E_{v_{lr}}, E_{v_{rl}}, E_{v_{rr}}$  as  $E_{ll}, E_{lr}, E_{rl}, E_{rr}$ , and  $\Gamma_{v_{ll}}, \Gamma_{v_{lr}}, \Gamma_{v_{rl}}, \Gamma_{v_{rr}}$  as  $\Gamma_{ll}, \Gamma_{lr}, \Gamma_{rl}, \Gamma_{rr}$ , respectively. Figure 3.13 illustrates how to put these drawings together. We state the algorithm precisely:

1. Rotate  $E_{lr}$  by  $90^\circ$  and place it anywhere on the grid.
2. Place  $t(E_{ll})$  at  $(x(t(E_{lr})) - 2, y(t(E_{lr})))$ . Now, perform a left lift of  $\Gamma_{ll}$ , so  $\Gamma_{ll}^L$  is the final drawing of  $T_{v_{ll}}$ .
3. Rotate  $E_{rl}$  by  $-90^\circ$  and place  $t(E_{rl})$  at  $(x(t(E_{lr})) + 2 \cdot h_{max} + 2, y(t(E_{lr})))$ . Observe that the base edges of  $E_{lr}$  and  $E_{rl}$  are horizontally two units apart.
4. Place  $t(E_{rr})$  at  $(x(t(E_{rl})) + 1, y(t(E_{rl}) - 1))$ .
5. Place  $\rho.\text{left}$  at  $(x(t(E_{lr})) - 1, y(t(E_{lr})))$ .
6. Place  $\rho.\text{right}$  at  $(x(t(E_{rl})) + 1, y(t(E_{rl})))$ .
7. To draw the root vertex  $\rho$ , let  $l_1$  be the line containing  $\rho.\text{left}$  with slope  $\pi/4$  and let  $l_2$  be the line containing  $\rho.\text{right}$  with slope  $-\pi/4$ . Place  $\rho$  at the intersection of  $l_1$  and  $l_2$  and draw edges  $(\rho, \rho.\text{left})$  and  $(\rho, \rho.\text{right})$ . Observe that  $\rho$  is placed on a grid point.
8. Finally, insert the edges  $(\rho.\text{left}, v_{ll}), (\rho.\text{left}, v_{lr})$  and  $(\rho.\text{right}, v_{rl}), (\rho.\text{right}, v_{rr})$  which completes the drawing.

Let  $\Gamma$  be the resulting straight-line drawing. We must argue that it satisfies all the properties.

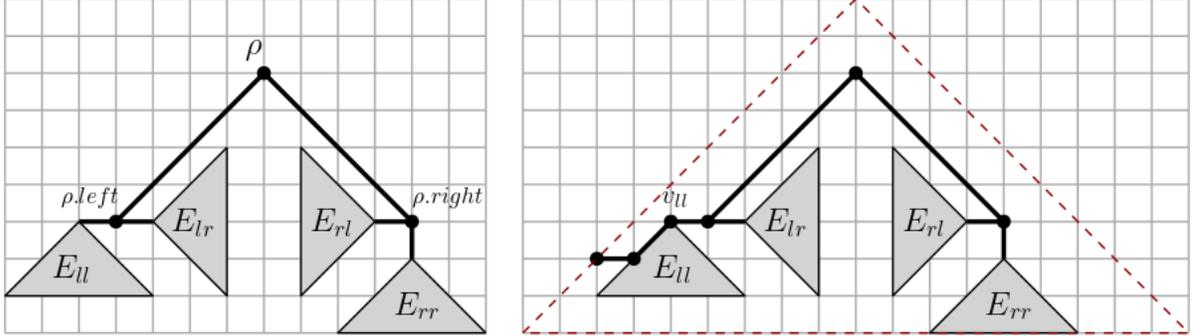


Figure 3.13: Left: placement of  $E_{ll}, E_{lr}, E_{rl}, E_{rr}$  before the lift on  $\Gamma_{ll}$  was performed. Right: the final drawing of  $\Gamma$  and  $Enc(\Gamma)$  (dashed).

$\Gamma$  is **planar and embedding-preserving**. Since  $\Gamma_{ll}^L, \Gamma_{lr}, \Gamma_{rl}$  and  $\Gamma_{rr}$  are planar and embedding-preserving by the induction hypothesis and the left lift invariant, it is sufficient to show that their respective enclosing triangles do not pairwise intersect and that  $\rho, \rho.left$  and  $\rho.right$  and their incident edges do not create any crossings. First, since all of  $E_{ll}, E_{lr}, E_{rl}$  and  $E_{rr}$  have equal dimensions by construction and since their enclosing triangles are isosceles by definition, by our relative placement of their top points, it is clear that they do not create any crossings. Drawings  $\Gamma_{lr}, \Gamma_{rl}$  and  $\Gamma_{rr}$  are contained within their enclosing triangles, so they cannot pairwise intersect either. For  $\Gamma_{ll}^L$ , by property (L4) of the left lift invariant, we know that  $R(\Gamma_{ll})$  remains within  $E_{ll}$ , so it cannot create crossings with  $\Gamma$ . Also, observe that by our placement of sub-tree drawings, no vertex is placed above the leftmost path  $L(\Gamma_{ll})$ , therefore  $\Gamma_{ll}^L$  does not intersect the other sub-tree drawings either.

Now, observe that  $\rho.left$  and  $\rho.right$  are placed one unit left of  $t(E_{lr})$  and one unit right of  $t(E_{rl})$  respectively, and consequently  $y(\rho) = y(b_r(E_{lr})) + 2$  and  $x(\rho) = x(b_r(E_{lr})) + 1$ . It is therefore clear that edges  $(\rho, \rho.left)$  and  $(\rho, \rho.right)$  do not intersect with  $\Gamma$ . Further, since  $v_{ll}$  and  $v_{rr}$  have the biggest  $y$  coordinates in  $\Gamma_{ll}^L$  and  $\Gamma_{rr}$ , it follows that  $(\rho.left, v_{ll})$  and  $(\rho.right, v_{rr})$  cannot create any crossings. Similarly,  $v_{lr}$  and  $v_{rl}$  have smallest and biggest  $x$  coordinates in  $\Gamma_{lr}$  and  $\Gamma_{rl}$  respectively as they are rotated versions of staircase drawings, hence  $(\rho.left, v_{lr})$  and  $(\rho.right, v_{rl})$  cannot create any crossings either. Lastly,  $\Gamma$  is clearly order preserving by our placement of  $E_{ll}, E_{lr}, E_{rl}$  and  $E_{rr}$ .

$\Gamma$  is an **extreme HV drawing**. For  $L(\Gamma)$ , note that the edge  $(\rho, \rho.left)$  is drawn with slope  $\pi/4$  downward by construction and  $(\rho.left, v_{ll})$  is a unit-length horizontal segment because  $v_{ll}$  is placed at  $t(E_{ll})$  after the left lift. Then, by Invariant 3,  $L(\Gamma_{ll})$  is diagonal-

horizontal. By property (L2) of the left lift invariant,  $L(\Gamma_{ll}^L)$  remains diagonal horizontal; therefore  $L(\Gamma)$  is diagonal horizontal. For  $R(\Gamma)$ , again observe that  $(\rho, \rho.\text{right})$  is drawn downward with slope  $-\pi/4$  by construction and that  $R(\Gamma_{rr})$  is diagonal-vertical by Invariant 1. To complete the argument, it suffices to note that by the definition of enclosing triangle we know  $x(v_{rr}) = x(t(E_{rr}))$ , so  $(\rho.\text{right}, v_{rr})$  is a vertical line segment. Recall that vertical segments are not required to be unit-length.

For the *extreme* property, by induction and by property (L6) of the left lift invariant, it is easy to see that all vertices of  $\Gamma_{ll}^L$  are in the trapped region of  $\Gamma$ . The same can be argued for the vertices of  $\Gamma_{rr}$ , since  $\Gamma_{rr}$  is an extreme HV drawing. The statement is clear for the vertices of  $\Gamma_{lr}$  and  $\Gamma_{rl}$ . See Figure 3.13 (right).

We now show  $\Gamma$  is a weakly zig-zag star-shaped drawing, and recall that the pole recovery conditions, (SS3) and (ZZS3), are not required to hold.

**$\Gamma$  is a star-shaped drawing, (ZZS1).** By induction and the left lift invariant, we know that  $\Gamma_{ll}^L, \Gamma_{lr}, \Gamma_{rl}$  and  $\Gamma_{rr}$  are star-shaped. It remains to show that the left and right polygons of  $\rho$ ,  $\rho.\text{left}$  and  $\rho.\text{right}$  satisfy conditions (SS1) and (SS2). See Figure 3.14 (left) for an illustration of this proof.

For vertex  $\rho$ , observe that since  $\rho$  is placed at  $(x(b_r(E_{lr})) + 1, y(b_r(E_{lr})) + 2)$ , it belongs to the SE region of the (rotated)  $E_{lr}$  and the SW region of the (rotated)  $E_{rl}$ . Since  $R(\Gamma_{lr})$  is drawn diagonal-vertical, by Invariant 1 and condition (V1) of Lemma 6, it follows that  $\rho$  sees every vertex on  $R(\Gamma_{lr})$ . Similarly, since  $L(\Gamma_{rl})$  is drawn diagonal-vertical, by Invariant 2 and condition (V2) of Lemma 6, it follows that  $\rho$  sees every vertex of  $L(\Gamma_{rl})$ , therefore (SS1) holds for the vertex  $\rho$ . Further observe that a vertical line going through point  $\rho$  separates  $P_l(\rho)$  and  $P_r(\rho)$ , hence (SS2) holds for  $\rho$  as well, i.e., polygons  $P_l(\rho)$  and  $P_r(\rho)$  do not contain vertices their interior.

We now show (SS1) and (SS2) for the vertex  $\rho.\text{left}$ . Since  $\rho.\text{left}$  is placed one unit horizontally to the right of  $v_{ll}$ , it belongs to the NE region of  $E_{ll}$ , hence by property (L3) of the left lift invariant,  $\rho.\text{left}$  sees  $R(\Gamma_{ll}^L)$ . Also, by the definition of enclosing triangle, we know that  $v_{lr}$  and  $t(E_{lr})$  have the same  $y$  coordinate ( $E_{lr}$  is rotated), so  $\rho.\text{left}$  is placed in the NW region of  $E_{lr}$ . Therefore by Invariant 1 and condition (V4) of Lemma 6,  $\rho.\text{left}$  sees every vertex on  $L(\Gamma_{lr})$ , so (SS1) holds for  $\rho.\text{left}$ . Also, observe that  $P_l(\rho.\text{left})$  and  $P_r(\rho.\text{left})$  are separated by a line with slope  $-\pi/4$  going through  $\rho.\text{left}$ , so (SS2) holds for  $\rho.\text{left}$ .

Finally, we show (SS1) and (SS2) for the vertex  $\rho.\text{right}$ . Since  $\rho.\text{right}$  is placed one unit vertically above  $v_{rr}$ , it belongs to the NW region of  $E_{rr}$ , hence by Invariant 3 and condition (V4) of Lemma 6,  $\rho.\text{right}$  sees  $L(\Gamma_{rr})$ . By the definition of the enclosing triangle, we know

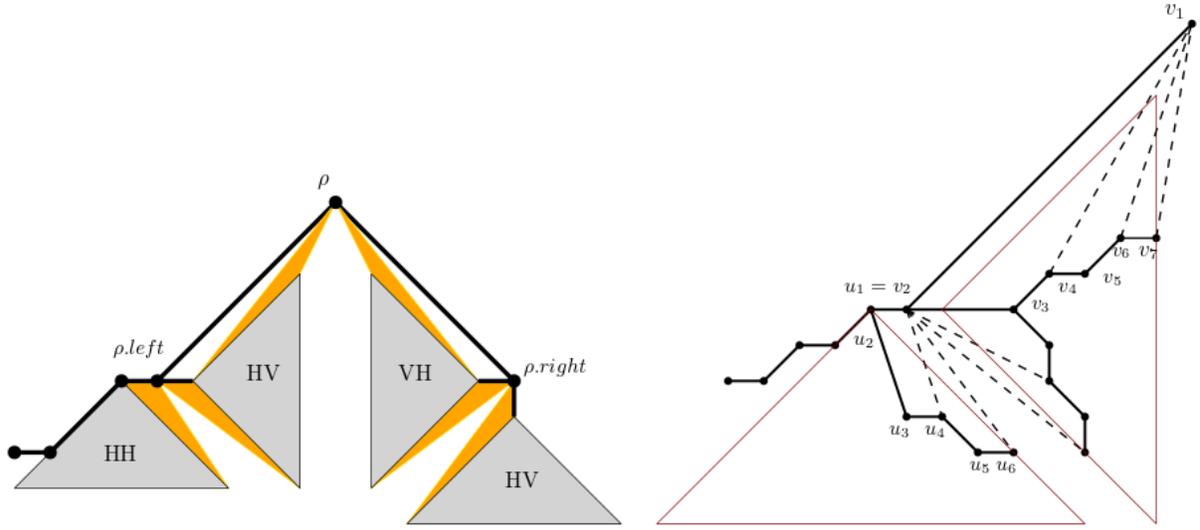


Figure 3.14: Left: Illustration that  $\Gamma$  is star-shaped. Right: Illustration of property (ZZS2) for  $\rho$  and  $\rho.left$ . Note that the lengths of the edges are not to scale.

that  $v_{rl}$  and  $t(E_{rl})$  have the same  $y$  coordinate. Hence  $\rho.right$  is placed in the NE region of  $E_{rl}$ , and therefore by Invariant 2 and condition (V3) of Lemma 6,  $\rho.right$  sees every vertex on  $R(\Gamma_{rl})$ , so (SS1) holds for  $\rho.right$ . Since  $P_l(\rho.right)$  and  $P_r(\rho.right)$  are separated by a line with slope  $\pi/4$  going through  $\rho.right$ , (SS2) holds for  $\rho.right$  as well.

$\Gamma$  admits the “zig-zag property” (ZZS2). First note that by induction and property (L1) of the left lift invariant, the zig-zag property holds for drawings  $\Gamma_{ll}^L, \Gamma_{lr}, \Gamma_{rl}$  and  $\Gamma_{rr}$ . It remains to prove condition (ZZS2) for  $\rho$ ,  $\rho.left$  and  $\rho.right$ . See Figure 3.14 (right) for an illustration of this proof.

So, let  $v_1, v_2, \dots, v_k$  be the left-right path of  $\rho$  and note that  $v_1 = \rho, v_2 = \rho.left$  and  $v_3 = v_{lr}$ . Since  $E_{lr}$  is an  $HV$  drawing that was rotated by  $90^\circ$ , edge  $(v_{lr}, v_{lr}.right)$  is drawn diagonal with slope  $\pi/4$  upward. Therefore, since edge  $(\rho.left, v_{lr})$  is drawn horizontal,  $\rho.left$  sees  $v_{lr}.right$ , so  $v_1, v_2, v_3, v_4$  is a strictly convex quadrangle. Further, since  $R(\Gamma_{lr}) := v_3, v_4, \dots, v_k$  is a diagonal-vertical path, observe that for all even  $i$ ,  $4 \leq i \leq k - 2$ ,  $v_i$  sees  $v_{i+2}$ , so  $v_1, v_i, v_{i+1}, v_{i+2}$  is a strictly convex polygon. A symmetric argument holds for the right-left path of  $\rho$ , so property (ZZS2) holds for vertex  $\rho$ .

For vertex  $\rho.left$  ( $u_1$  in Figure 3.14 (right)), let  $R(\Gamma_{ll}^L) = u_2, u_3, \dots, u_t$  ( $v_{ll} = u_2$ ) be the rightmost path of  $\Gamma_{ll}^L$ . By condition (L5) of the left lift invariant, we know that  $u_i$  sees  $u_{i+2}$

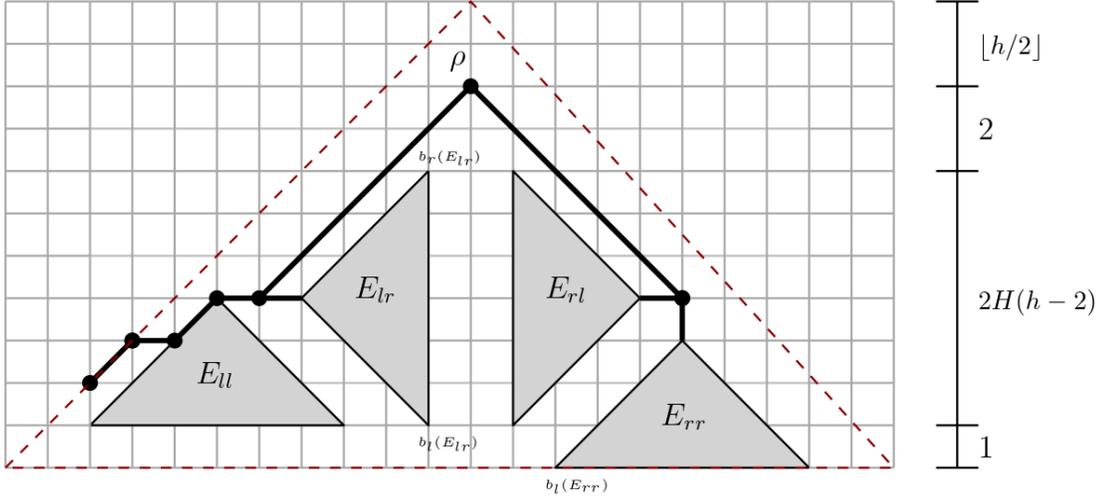


Figure 3.15: Illustration of the grid size analysis.

for all even  $i$  with  $2 \leq i \leq t - 2$ . Therefore, property (ZZS2) holds for the left-right path  $u_1, u_2, \dots, u_t$ . Since the leftmost path of  $\Gamma_{lr}$  is also diagonal-horizontal by Invariant 2, by a similar argument it follows that the right-left path of  $\rho$ .left also satisfies (ZZS2). Similar arguments hold for  $\rho$ .right.

**Grid size of  $\Gamma$ .** Finally, we give an upper bound on the height of  $Enc(\Gamma)$ . For a tree of height  $h$ , let  $H_i(h)$  be the height of the enclosing triangle of the drawing satisfying Invariant  $i$  for  $i \in \{1, 2, 3\}$  and let  $H(h) := \max\{H_1(h), H_2(h), H_3(h)\}$ . The reason for taking the maximum of the heights is that we enlarged the original enclosing triangles to get  $E_{ll}, E_{lr}, E_{rl}$  and  $E_{rr}$ . We need to show that  $H(h) \leq 10 \cdot 2^{\lfloor h/2 \rfloor} - h - 7$ .

Recall that the base cases are already verified and that  $h \geq 2$ . Now, since the width of the enclosing triangle is twice its height, root  $\rho$  is placed  $2H(h - 2) + 3$  units above the bottom of  $Enc(\Gamma)$ . Observe that  $\rho$  is placed 2 units above  $b_r(E_{lr})$  and that  $b_l(E_{rr})$  is drawn 1 unit below  $b_l(E_{lr})$ , see Figure 3.15. Since the number of edges on the diagonal-horizontal path is  $h$  (the height of the tree), by Lemma 5, the top of the  $Enc(\Gamma)$  is  $\lfloor h/2 \rfloor$  units above the root  $\rho$ . Therefore the height of the  $Enc(\Gamma)$  is

$$H_1(h) = 2H(h - 2) + 3 + \lfloor h/2 \rfloor$$

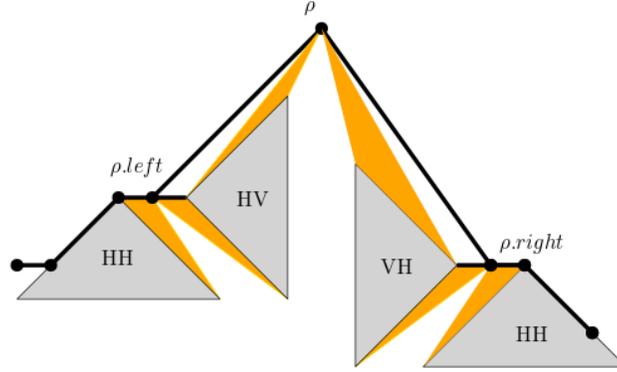


Figure 3.16: Illustration that vertex  $\rho$  still sees the vertices of  $L(E_{rl})$  after the left lift.

We now show that  $H_1(h) \leq 10 \cdot 2^{\lfloor h/2 \rfloor} - h - 7$ :

$$\begin{aligned}
H_1(h) &= 2 \cdot H(h-2) + 3 + \lfloor h/2 \rfloor \\
&\leq 2 \cdot (10 \cdot 2^{\lfloor (h-2)/2 \rfloor} - (h-2) - 7) + 3 + \lfloor h/2 \rfloor \\
&= 2 \cdot (10 \cdot 2^{\lfloor h/2 - 1 \rfloor} - (h-2) - 7) + 3 + \lfloor h/2 \rfloor \\
&= 10 \cdot 2^{\lfloor h/2 \rfloor} - 2h + 4 - 14 + 3 + \lfloor h/2 \rfloor \\
&= 10 \cdot 2^{\lfloor h/2 \rfloor} - h - 7 - (h - \lfloor h/2 \rfloor + 7 - 7) \\
&\leq 10 \cdot 2^{\lfloor h/2 \rfloor} - h - 7.
\end{aligned}$$

Here, the second inequality follows by the inductive hypothesis and the sixth inequality follows since  $(h - \lfloor h/2 \rfloor + 7 - 7) > 0$ .

*Constructions 2 and 3.*

See Figure 3.17 for how to put together the sub-tree drawings for Invariants 2 and 3. The VH drawings of Invariant 2 are symmetric to those of Invariant 1. To obtain HH drawings for Invariant 3, we recursively draw both  $E_{ll}$  and  $E_{rr}$  as HH drawings (Invariant 3). In Invariant 2, we apply the lifting operation on  $E_{rr}$  and in Invariant 3 we lift both  $E_{ll}$  and  $E_{rr}$ . In addition, HH drawings of Invariant 3 are required to satisfy the lift invariants. Here we show that the left lift invariant holds.

**$\Gamma$  satisfies the left lift invariant.** Recall that the left lift  $\Gamma^L$  of  $\Gamma$  is obtained by shifting the root vertex  $\rho$  vertically, along with its left sub-tree, so that  $\rho$  is placed at the top of  $Enc(\Gamma)$ , see Figure 3.12. Observe that the drawings of  $T_{\rho, \text{left}}$  and  $T_{\rho, \text{right}}$  are separated by

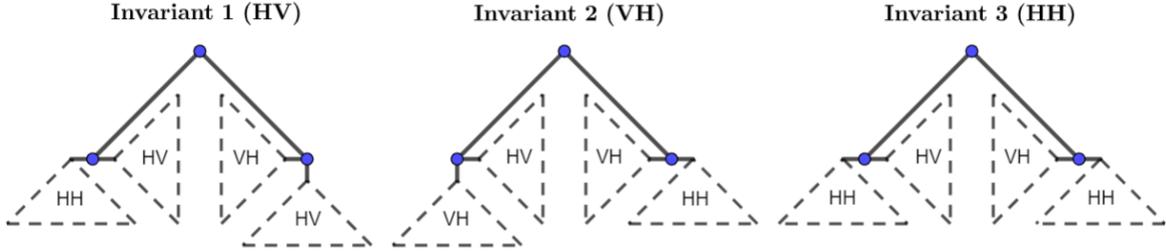


Figure 3.17: Placement of the sub-tree drawings for each invariant before the lifts are performed. In Invariant 2, we perform the right lift on  $\Gamma_{rr}$ , the HH drawing. In Invariant 3, we perform the left lift on  $\Gamma_{ll}$  and the right lift on  $\Gamma_{rr}$ .

the vertical line containing the root vertex  $\rho$ ; therefore planarity of  $\Gamma^L$  follows immediately. For property (L1), the key thing to show is that after the left lift, the root vertex  $\rho$  still sees all the vertices on  $L(E_{rl})$ . To see this, it is sufficient to observe that after the upward vertical lift,  $\rho$  still remains in the SW region of the (rotated)  $E_{rl}$ , see Figure 3.16. The remaining left lift invariant properties of  $\Gamma^L$  are easily verified.

The arguments for the remaining properties are almost identical to the ones presented above for the HV drawings, so we omit the proof.  $\square$

### Step 5: Drawing Outer-1-Planar Graphs

We now make use of the zigzag star-shaped binary tree drawings to draw any quadrangular outer-1-plane graph:

**Theorem 16** *Let  $G_{st}$  be a quadrangular  $o1p$  graph and let  $h_t$  be the height of the dual ternary tree of  $\text{skel}(G)$ . Then  $G$  admits an outer-1-plane straight-line drawing  $\Gamma$  so that both the height and the width of  $\Gamma$  are  $O(2^{h_t})$ .*

*Proof.* First, compute the half-skeleton  $M(G)$ , let  $T$  be its dual binary tree rooted at the face containing edge  $(s, t)$  and let  $h_b$  be its height. By Theorem 15, produce a weak zigzag star-shaped HV drawing  $\Gamma_T$  of  $T$  so that both the height and the width of its enclosing triangle are  $O(2^{h_b/2})$ . We now show that  $\Gamma_T$  also satisfies condition (ZZS3) of Definition 4. We define point  $p_s$  to be  $(x(t(\text{Enc}(\Gamma_{(T_M)}))), y(t(\text{Enc}(\Gamma_{(T_M)}))) + 1)$  and  $p_t$  to be  $(x(b_r(\text{Enc}(\Gamma_{(T_M)}))) + 1, y(b_r(\text{Enc}(\Gamma_{(T_M)}))))$ . By Lemma 6,  $p_s$  sees all vertices on the leftmost path of  $\Gamma_T$  and  $p_t$  sees all the vertices on the rightmost path of  $\Gamma_T$ . Since  $\Gamma_T$  is an HV drawing, it is easy to see that  $p_s$  and  $p_t$  satisfy all three properties of condition

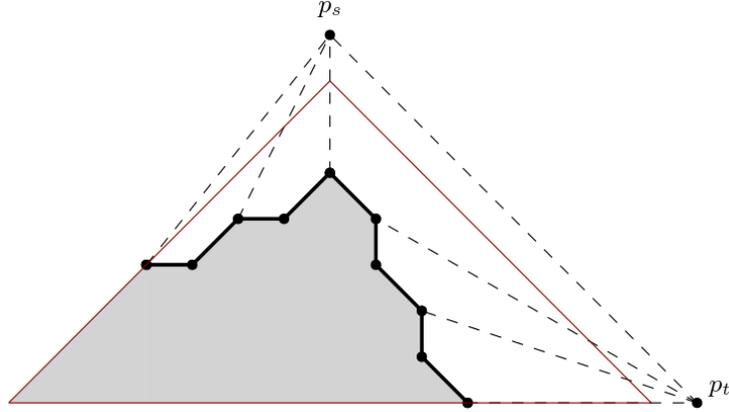


Figure 3.18: Demonstration that *HV* drawings satisfy property (ZZS3).

(ZZS3), see Figure 3.18 for an illustration. Also, note that the height and the width of the bounding box of  $\Gamma_T \cup \{p_s, p_t\}$  are both only one unit greater than the height and the width of  $Enc(\Gamma_T)$ . Therefore, by Corollary 3, graph  $G$  admits an outer-1-plane straight line drawing  $\Gamma$  in  $O(2^{h_b/2}) \cdot O(2^{h_b/2})$  area.

Now, since  $skel(G)$  consists of quadrangles only, it follows that  $h_b \leq 2 \cdot h_t + 1$ ; triangulating  $skel(G)$  can at most double the height of its dual tree. Therefore, by substituting  $2 \cdot h_t + 1$  in  $h_b$ , we have that  $O(2^{h_b/2}) \subseteq O(2^{(2h_t+1)/2}) = O(2^{h_t})$ , so the theorem follows.  $\square$

Finally, consider the special case of complete o1p graphs, i.e., o1p graphs whose dual ternary tree is complete.

**Corollary 4** *Every  $n$ -vertex complete quadrangular o1p graph  $G$  admits an outer-1-plane straight-line drawing  $\Gamma$  so that both the height and the width of  $\Gamma$  are  $O(n^{0.63})$ ; so the area is  $O(n^{1.26})$ .*

*Proof.* Let  $h_t$  be the height of the dual tree  $T$  of  $skel(G)$ . Since  $|T| \leq n$  and since the height of any  $k$ -node ternary tree is  $\lfloor \log_3(2k) \rfloor$ , it follows that  $h_t \leq \lfloor \log_3(2n) \rfloor$ . By Theorem 16, let  $\Gamma$  be the outer-1-plane straight-line drawing of  $G$  so that both the height and the width of  $\Gamma$  are  $O(2^{h_t})$ . Therefore, substituting  $\lfloor \log_3(2n) \rfloor$  for  $h_t$  and since  $z^{\log_a(b)} = b^{\log_a(z)}$ , we have that  $O(2^{\lfloor \log_3(2n) \rfloor}) \subseteq O(2^{\log_3(2n)}) = O((2n)^{\log_3(2)}) \approx O(n^{0.63})$ .  $\square$

Observe that Theorem 15 gives linear area drawings of complete *binary trees*. However, complete o1p graphs are defined as those with complete *ternary tree* duals, which is why, after doubling the height of the dual tree in Theorem 11, we lost the linear area bound. To

reduce the area to linear, one would need to use a quite different placement of the subtrees and possibly go “deeper” than the grand-children.

There are graph drawing algorithms for trees that are somewhat balanced, yet not complete. For instance, Crescenzi, Di Battista and Piperno [10] gave linear area drawings for Fibonacci trees and more generally, Crescenzi, Penna and Piperno [11] gave linear area drawings for AVL trees. However, the resulting drawings are not embedding-preserving and we do not see a way to make them star-shaped, so their drawing techniques do not seem applicable in our case. Another idea to work towards linear area drawings of complete  $o1p$  graphs would be to forgo the detour into drawings binary trees altogether and to draw them directly. Still, the problem of finding linear area drawings of complete  $o1p$  graphs remains open.

# Chapter 4

## Orthogonal Drawings of Outer-1-Plane Graphs

In this chapter, we show that every outer-1-plane graph with maximum degree 4 admits an embedding-preserving orthogonal point-drawing in  $O(n \log n)$  area with at most 7 bends per edge. Our method is a modification of Biedl's algorithm [5] which finds a flat orthogonal box-drawing of any outer-planar graph in  $O(n \log n)$  area.

### 4.1 Review of Existing Results

Recall that orthogonal point-drawings are poly-line grid drawings, with an additional constraint that every line-segment of a poly-line must either be horizontal or vertical, see Figure 4.1 (right). Orthogonal drawings gained popularity because of their wide applicability to floor-planning, VLSI design, software architecture and more [4]. Besides minimizing the area of orthogonal drawings, a common goal is to minimize the number of bends, whether the total number of bends or the maximum number of bends per edge. In this chapter, we focus on minimizing the maximum number of bends per edge. Because of the nature of orthogonal drawings, we only consider graphs of maximum degree four. Observe that every vertex has four available *ports* for edges: the north, south, west and east port.

We first review the results for outer-planar graphs. In 1981, Dolev and Trickey [17] showed that every outer-planar graph admits a planar orthogonal drawing in  $O(n)$  area with at most  $O(\log n)$  bends per edge. The key ingredient to their method is Valiant's linear area orthogonal drawing algorithm for trees [38]. Since weak duals of outerplanar

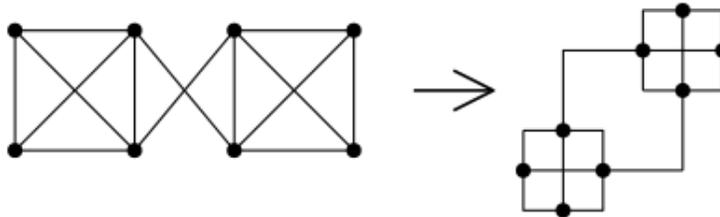


Figure 4.1: (Argyriou et al. [2]) (Left) An example of a biconnected outer-1-plane graph which does *not* admit an embedding-preserving orthogonal drawing so that every edge has at most one bend. (Right) A non-embedding-preserving drawing of the same graph with at most one bend per edge; the middle crossing is not preserved.

graphs are trees, they first produce a drawing of the dual tree using Valiant’s algorithm and then recover the outer-planar graph while keeping the area small. As their algorithm is clearly optimal with respect to the area of the drawing, it is natural to attempt to minimize the number of bends per edge while keeping the area small. This was achieved by Tayu, Nomura and Ueno [36] who showed that any outer-planar graphs admits an orthogonal drawing with at most 1 bend per edge in  $O(n^2)$  area. The number of bends cannot be reduced to zero since a triangle has no straight-line orthogonal drawing.

Now we turn to outer-1-planar graphs. Recall that *flat orthogonal box-drawings* are drawings where vertices are drawn as horizontal line segments and every edge is a sequence of horizontal and vertical line segments. As will be explained in Section 4.4, flat orthogonal box-drawings are related to orthogonal point-drawings. Auer et al. [3] claimed an algorithm that produces a planar flat orthogonal box-drawing of any o1p graph in  $O(n \log n)$  area with no bends. However, Biedl [7] found a mistake in the paper and proved that there exists outer-1-planar graphs that require  $\Omega(n^2)$  area in any poly-line drawing without crossings.

The motivation for our result is the paper by Argyriou et al. [2], who showed that any 1-planar (and hence o1p) graph admits an embedding-preserving orthogonal drawing in  $O(n^2)$  area such that every edge has at most three bends. For biconnected o1p graphs, they improved the number of bends per edge to two. This algorithm achieves the minimum number of bends per edge possible, as there are biconnected o1p graphs so that in any embedding-preserving orthogonal drawing, at least one edge must have at least two bends [2]. See also Figure 4.1.

We focus on reducing the area of the drawing for o1p graphs to  $O(n \log n)$  while keeping the number of bends per edge relatively small. We begin the chapter by reviewing Biedl’s algorithm [5] for producing flat orthogonal box-drawings of outer-plane graphs in  $O(n \log n)$

area. We then extend the algorithm to work on outer-1-plane graphs and carefully convert it to an orthogonal point-drawing by adding not too many bends.

## 4.2 Review of Biedl’s Algorithm [5]

In this section we give a review of Biedl’s algorithm that produces a flat orthogonal box-drawing of any outer-planar graph in  $O(n \log n)$  area [5]. Note that the presentation here is almost identical to the original paper of Biedl.

**Theorem 17** *Let  $G$  be a maximal outer-plane graph. Then  $G$  admits an embedding-preserving orthogonal-box drawing  $\Gamma$  in  $O(n \log n)$  area such that every edge has at most 2 bends.*

*Proof.* Let  $(u, v)$  be an edge on the outer-face of  $G$  with  $u$  before  $v$  in the clockwise order on the outer-face; we call  $(u, v)$  the *reference edge*. We prove the following invariant:

- $u$  occupies the top right corner of the drawing,
- $v$  occupies the bottom right corner of the drawing,
- edges that attach horizontally to any box have no bends,
- the height is at most  $3 \log n - 1$ ,
- the width is  $\frac{5}{2}n - 4$ , and
- every edge has at most 2 bends.

*Base Case.*  $n = 2$ .

Then simply place  $u$  on top of  $v$  in a single column. All invariant properties clearly hold.

*Induction Step.*

Let  $w$  be the third vertex of the triangle containing edge  $(u, v)$ . Further, let  $G_1$  be the subgraph induced by the vertices in between  $w$  and  $u$  in the clockwise order on the outer-face and similarly let  $G_2$  be the subgraph induced by vertices in between  $v$  and  $w$ , see Figure 4.2 (left). Assume that  $|V(G_1)| \leq |V(G_2)|$ , the other case is symmetric as will be shown later. We have two cases:

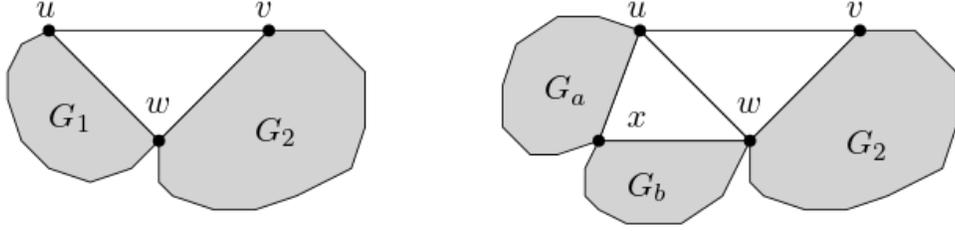


Figure 4.2: Left: Illustration of  $G_1$  and  $G_2$ . Right: Further splitting  $G_1$  in Case 2.

**Case 1.**  $|G_1| = 2$ , i.e.,  $u$  and  $w$  are the only vertices of  $G_1$ .

Then draw  $G_2$  recursively using edge  $(w, v)$  as the reference edge, place vertex  $u$  to the right of vertex  $w$  and draw edge  $(u, w)$  horizontally. Since by induction vertex  $v$  of  $G_2$  is placed on the bottom-right corner of the drawing of  $G_2$ , its bar can be simply extended to match the  $x$ -coordinate of  $u$ . The height does not increase, so is at most  $3 \log(|G_2|) - 1 \leq 3 \log n - 1$ , and the width is at most  $\frac{5}{2}|G_2| - 4 + 1 \leq \frac{5}{2}(n - 1) - 3 \leq \frac{5}{2}n - 4$  since we added one column. See Figure 4.3.

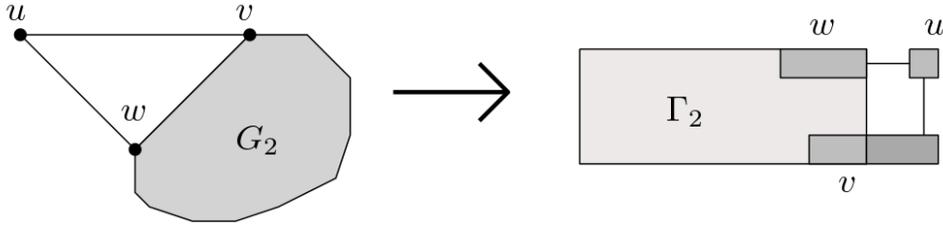


Figure 4.3: Illustration of Case 1.

**Case 2.**  $|G_1| \geq 3$ .

Then let  $x$  be the third vertex of the triangle of  $G_1$  containing the edge  $(u, w)$ . Now, let  $G_a$  and  $G_b$  be the subgraphs induced by the vertices between  $u$  and  $x$ , and  $x$  and  $w$  respectively, see Figure 4.2 (right). Draw  $G_a, G_b$  and  $G_2$  recursively using  $(u, x), (x, w)$  and  $(w, v)$  as reference edges to obtain the drawings  $\Gamma_a, \Gamma_b$  and  $\Gamma_2$ .

For purpose of merging the recursive drawings, we first slightly modify drawings  $\Gamma_a$  and  $\Gamma_b$ . Note that by induction vertex  $x$  occupies the bottom right corner of drawing  $\Gamma_a$ . Now, re-allocate vertex  $x$  so it “spans” the bottom row of  $\Gamma_a$  by moving it down to a new row and extending it to match the width of drawing  $\Gamma_a$ . All vertical edges incident to  $x$  can be extended to reach  $x$  in a new row. Since horizontal edges have no bends by induction, they will now become vertical and connect to  $x$  in the bottom row. See Figure 4.4 for an

illustration. Let  $\Gamma'_a$  be the resulting modified drawing. Similarly let  $\Gamma'_b$  be modification of drawing  $\Gamma_b$  obtained by moving  $x$  to the top row.

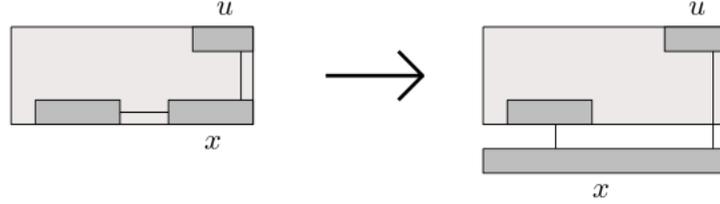


Figure 4.4: Releasing vertex  $x$  so it spans the bottom row of the drawing.

Finally, we merge  $\Gamma'_a, \Gamma'_b$  and  $\Gamma_2$  to obtain a drawing of the graph  $G$ . Add empty rows to  $\Gamma'_a$  or  $\Gamma'_b$  until both drawings have the same height. Similarly, stretch  $\Gamma_2$  so its height is two rows bigger than  $\Gamma'_a$  and  $\Gamma'_b$ , without exceeding the height bound of  $3 \log n - 1$ . This can be done, as will be argued in the analysis of the height. Now, place all three drawings next to each other as shown in Figure 4.5;  $\Gamma'_b$  is rotated. The two boxes of  $w$  can be joined into one box as they are in the same row next to each other. Since the two boxes of  $x$  span the bottom row of  $\Gamma'_a$  and  $\Gamma'_b$ , they can similarly be merged into one box. Finally, since  $\Gamma_2$  is two rows taller than  $\Gamma'_b$  (and  $\Gamma'_a$ ), this leaves us with enough space to extend the vertex  $v$  to the bottom right corner of the drawing and to insert edge  $(u, v)$  as shown in Figure 4.5.

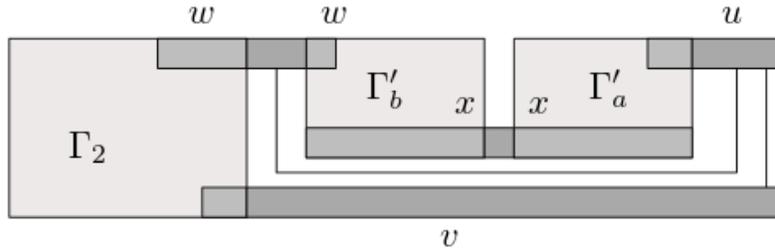


Figure 4.5: Illustration of combining drawings  $\Gamma'_a, \Gamma'_b$  and  $\Gamma_2$  in Case 2.

For the height bound, since  $|G_1| \leq |G_2|$  and they share only one vertex  $w$ , we have  $|G_1| \leq \frac{n+1}{2}$ . As  $G_a \subseteq G_1 \setminus \{w\}$ , it follows that  $|G_a| \leq \frac{n}{2}$ . By the inductive hypothesis, the height of  $\Gamma_a$  is  $3 \log |G_a| - 1 \leq 3 \log n - 4$ . After adding one extra row to  $\Gamma_a$  to move  $x$ , the height of drawing  $\Gamma'_a$  is  $3 \log n - 3$ . Lastly, after stretching  $\Gamma_2$ , the height of the final drawing is 2 units taller than  $\Gamma'_a$ , resulting in  $3 \log n - 1$  height, which satisfies the inductive hypothesis.

The width of the drawing is the sum of the widths of  $\Gamma_2, \Gamma_b$  and  $\Gamma_a$  plus three columns added. So, since  $|G_2| + |G_a| + |G_b| = n + 2$ , by induction it follows that the width is at most

$$3 + \frac{5}{2}|G_2| - 4 + \frac{5}{2}|G_a| - 4 + \frac{5}{2}|G_b| - 4 = 3 + \frac{5}{2}(n + 2) - 12 = \frac{5}{2}n - 4.$$

Lastly, when  $|G_1| \geq |G_2|$ , the construction is symmetric and omitted here. See Figure 4.6 for an illustration.  $\square$

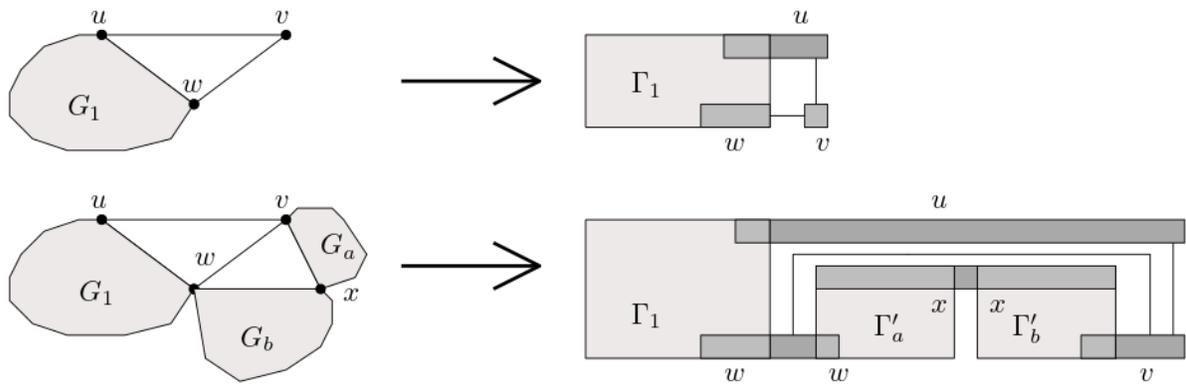


Figure 4.6: Case 1 and Case 2 when  $|G_1| \geq |G_2|$ .

### 4.3 Inserting the Missing Diagonals

We now extend Biedl’s algorithm to outer-1-planar graphs. Let  $G$  be a maximal outer-1-plane graph and let  $H$  be any half-skeleton of  $skel(G)$ . First, recall that an edge  $(x, y)$  is a *missing diagonal* of  $H$  if  $(x, y)$  is an edge of  $G \setminus H$ . This means that  $(x, y)$  was crossed in  $G$  and the edge that it crosses belongs to  $H$ . To draw  $G$ , we run Biedl’s algorithm on  $H$ , a maximal outer-plane graph, and modify the recursive step to insert the missing diagonals.

We treat any missing diagonal as two *half-edges*. For example, suppose that  $abc$  and  $acd$  are two adjacent triangles of  $H$  (say at edge  $(a, c)$ ), with vertices ordered in clockwise order. Suppose further that  $abcd$  is a quadrangle of  $skel(G)$ . When inserting the missing diagonal  $(b, d)$ , a half edge at  $b$  and a half-edge at  $d$  could be drawn separately towards some agreed point, e.g., a point on the edge  $(a, c)$  crossed by  $(b, d)$ , and then combined to form a full edge.

Consider any graph  $H'$  that arises as subgraph during the recursions to draw  $H$ , and let  $(u', v')$  be its reference edge. If  $|H'| \geq 3$ , then we use  $w(H')$  to denote the third vertex of the triangle of  $H'$  that contains  $(u', v')$ . There may be a missing diagonal  $e$  of  $G$  that crosses  $(u', v')$  and so necessarily ends at  $w(H')$ . Edge  $e$  is technically not a missing diagonal of  $H'$  (its other endpoint does not belong to  $H'$ ), but we call the half-edge of  $e$  between  $w(H')$  and the crossing-point the *connector half-edge of  $H'$*  and insert this half-edge when drawing  $H'$ . We now state our modification of Biedl’s algorithm:

**Theorem 18** *Let  $G$  be an outer-1-plane graph and let  $H$  be any half-skeleton of  $G$ . Then  $G$  admits a flat orthogonal box-drawing  $\Gamma$  in  $O(n \log n)$  area such that:*

1. *Every missing diagonal of  $H$  has at most 4 bends in  $\Gamma$ .*
2. *All the remaining edges have at most 2 bends in  $\Gamma$ .*

*Proof.* Let  $(u, v)$  be an arbitrary reference edge on the outer-face of  $H$  with  $u$  before  $v$  in the clockwise order on the outer-face. We prove a slightly stronger claim, namely, in addition to the invariant properties of Biedl, we show that

- the connector half-edge at  $w$ , if it exists, is drawn with 1 bend,
- all the missing diagonals of  $H$  are drawn with at most 4 bends, and
- the height is at most  $5 \log n - 3$ .

- edge  $(u, v)$  is drawn without bend vertically in the rightmost column.

In the base case, when  $n = 2$ , place  $v$  on top of  $u$  in a single column; so there is no change to Biedl's algorithm. In the inductive step, when  $n \geq 3$ , let  $w := w(H)$  and let  $H_1$  ( $H_2$ ) be the subgraph induced by the vertices between  $w$  and  $u$  ( $v$  and  $w$ ) on the outer-face. That is, we use  $H_1$  and  $H_2$  to denote what  $G_1$  and  $G_2$  stood for in Biedl's algorithm (we use  $H$  now to remind us that these are half-skeletons and not the full o1p graph  $G$ ). Also, assume that  $|V(H_1)| \leq |V(H_2)|$ , inserting the missing diagonals in the other case will be symmetric. Now, let  $G_i$  (for  $i = 1, 2$ ) be graph  $H_i$  with applicable missing diagonals added.

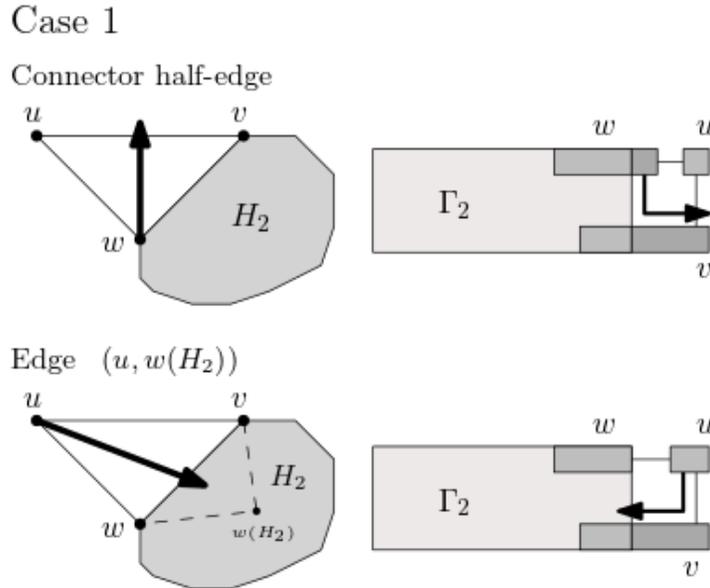


Figure 4.7: Demonstration of Case 1.

**Case 1.**  $|H_1| = 2$ , i.e.,  $u$  and  $w$  are the only vertices of  $H_1$ .

Draw  $G_2$  recursively to get  $\Gamma_2$ . To draw  $H$ , we extend  $\Gamma_2$  in the same way as in Biedl's algorithm (Theorem 17). The connector half-edge at  $w$  (if it exists) can be drawn with one bend as shown in Figure 4.7 (top), after adding a column to expand  $w$  and (if  $\Gamma_2$  has height 2) a row, without exceeding the  $5 \log n - 3$  height bound (since  $5 \log n - 3 \geq 5 \log 3 - 3$  by  $n \geq 3$ ).

For the second part of the invariant, all the missing diagonals of  $H_2$  are drawn recursively with at most 4 bends. The only possible remaining missing diagonal is  $(u, w(H_2))$ . In this case, the connector half-edge at  $H_2$  is drawn with one bend by induction, and the

half-edge at  $u$  can be drawn with one bend as described in Figure 4.7 (bottom). Observe that both half-edges can be aligned without introducing extra bends, because the half-edge at  $u$  can enter  $\Gamma_2$  anywhere along edge  $(w, v)$  (which is drawn rightmost and vertical in  $\Gamma_2$ ) and align with the other half-edge.

**Case 2.**  $|H_1| \geq 3$ .

We use  $H_a, H_b$  and  $H_2$  to denote what  $G_a, G_b$  and  $G_2$  stood for in Biedl's algorithm. Now, we use  $G_a, G_b$  and  $G_2$  to denote  $H_a, H_b$  and  $H_2$  with the applicable missing diagonals added. Let  $\Gamma_a, \Gamma_b$  and  $\Gamma_2$  be the drawings of  $G_a, G_b$  and  $G_2$  obtained recursively using the same reference edges as before. Put the drawings  $\Gamma_a, \Gamma_b$  and  $\Gamma_2$  together in the same way as in Biedl's algorithm. See also Figure 4.8.

Now, stretch  $\Gamma_2$  vertically so it is 4 rows taller than  $\Gamma'_a$  and  $\Gamma'_b$  (as opposed to 2 rows in Biedl's algorithm), which gives us 2 additional free rows to insert the missing diagonals. As will be argued in the analysis of the height, the height of  $\Gamma_2$  will not exceed  $5 \log n - 3$ . We may hence assume that both the row above and the row below the horizontal segment of  $(u, w)$  contain no horizontal segments below  $\Gamma'_b$  and  $\Gamma'_a$ , see Figure 4.9.

We slightly modify the drawing depending on which edges are the missing diagonals. First observe that the connector half-edge at  $w$  can be drawn with one bend as shown in Case 2A in Figure 4.8, after appropriately adding one extra column. It uses the free row below  $(u, w)$ . All the missing diagonals belonging entirely to  $H_a, H_b$  and  $H_2$  are drawn with at most 4 bends by induction. Now, we consider the missing diagonals with at least one half-edge belonging to triangle  $uvw$  or  $uwv$ . Observe that these could be the edges  $(v, x), (w, w(H_a)), (u, w(H_b)),$  and  $(u, w(H_2))$ .

*Case 2B.*  $(v, x)$  is a missing diagonal.

Then it can be simply drawn with 0 bends as shown in Figure 4.8.

*Case 2C.*  $(u, w(H_2))$  is a missing diagonal.

Then the half-edge at  $w(H_2)$  is drawn with 1 bend by induction. Now, insert the half-edge at  $u$  with 3 bends as described in Figure 4.8, after appropriately adding a single column and using the free row below  $(u, w)$ . Observe that both half-edges can be aligned without creating additional bends as we have the choice where to enter drawing  $\Gamma_2$  along edge  $(v, w)$ . Thus the missing diagonal is drawn with 4 bends.

*Case 2D.*  $(w, w(H_a))$  is a missing diagonal.

Then the half-edge at  $w(H_a)$  is drawn with 1 bend by induction. Now, insert the half-edge at  $w$  with at most 3 bends as described in Figure 4.8, after appropriately adding two extra columns and using the free row above  $(u, w)$ . Observe that the half-edges can be

## Case 2

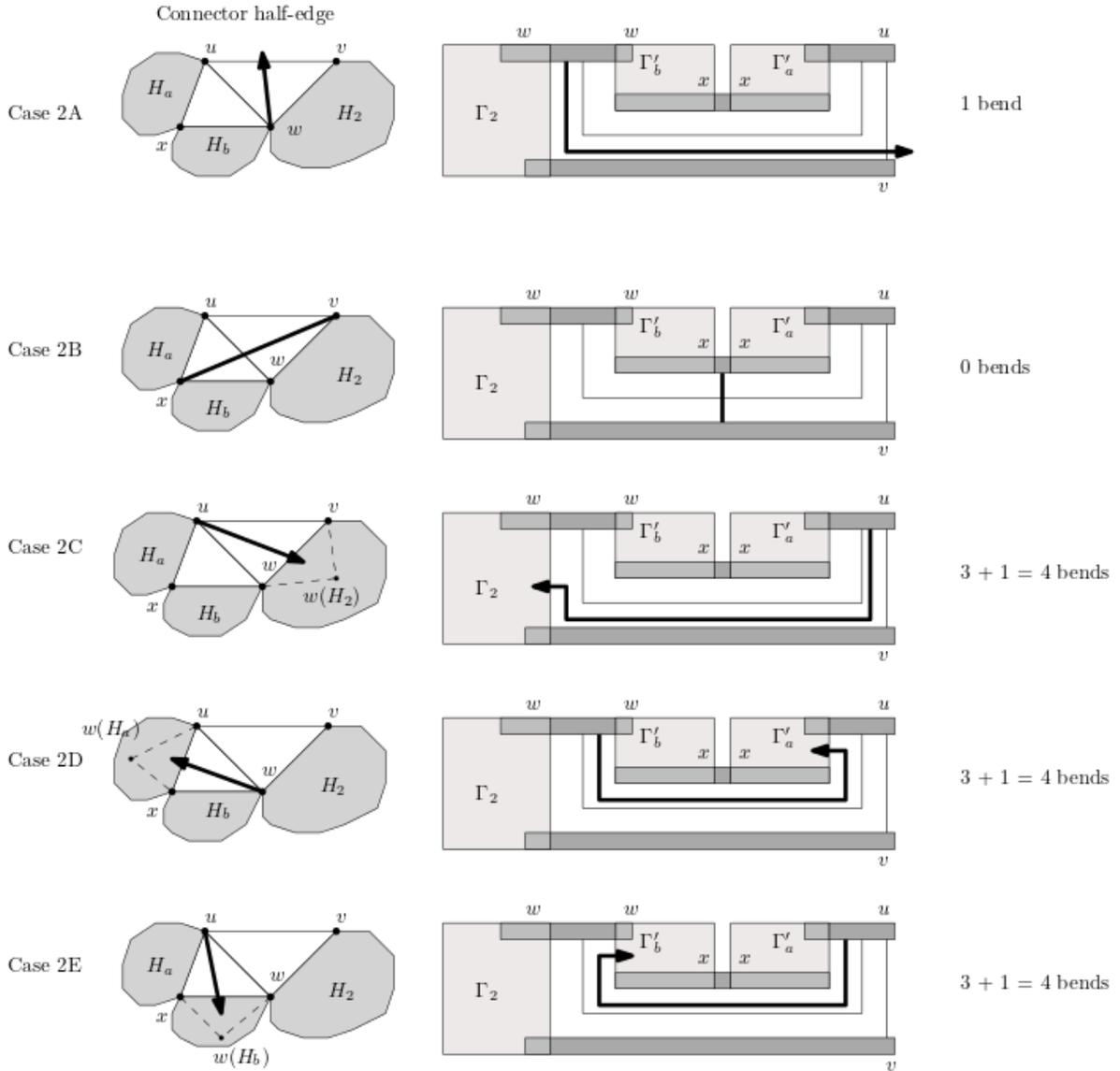


Figure 4.8: Demonstration of Case 2.

aligned without creating extra bends, as we have the choice where to enter drawing  $\Gamma_a$  along edge  $(v, x)$ . Thus the missing diagonal is drawn with 4 bends.

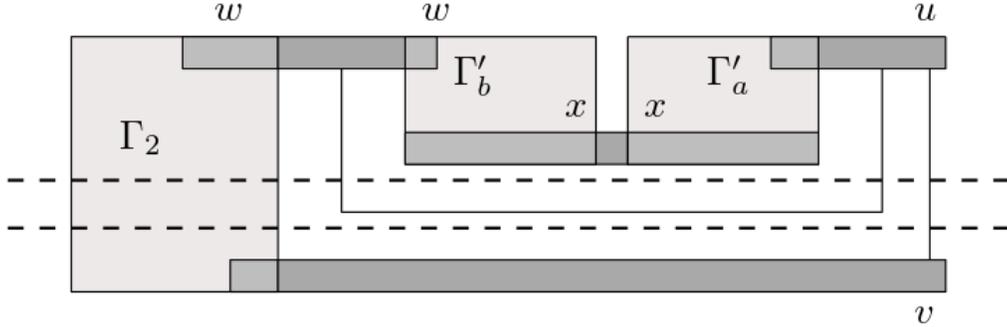


Figure 4.9: Illustration that the row above and the row below the horizontal line segment of  $(u, w)$  contain no horizontal line segments below  $\Gamma'_b$  and  $\Gamma'_a$ .

*Case 2E.*  $(u, w(H_b))$  is a missing diagonal.

This case is symmetric to Case 2D, see Figure 4.8.

Since  $u, w, w(H_b), x$  and  $u, w, x, w(H_a)$  cannot both be quadrangles of  $skel(G)$ , Case 2D and Case 2E cannot occur simultaneously. Likewise Case 2A and Case 2C cannot happen simultaneously. So each free row is used by at most one half-edge, and having two free rows is sufficient to simultaneously insert a half-edge in all cases.

For the height bound, we repeat the argument from Biedl's algorithm with a slight modification to account for two additional rows. In the base case ( $n = 2$ ), the drawing occupies two rows and  $5 \log 2 - 3 \geq 2$ . In Case 1 of the inductive step, we stretched  $\Gamma_2$  to at most  $5 \log n - 3$  height and already argued that this is sufficient to complete the drawing.

For Case 2, first note that since  $|H_1| \leq |H_2|$  and they share only one vertex  $w$ , we have  $|H_1| \leq \frac{n+1}{2}$ . As  $H_a \subseteq H_1 \setminus \{w\}$ , it follows that  $|H_a| \leq \frac{n}{2}$ . By the inductive hypotheses, the height of  $\Gamma_a$  is  $5 \log |H_a| - 3 \leq 5 \log n - 8$ . After adding one extra row to  $\Gamma_a$  to move  $x$ , the height of drawing  $\Gamma'_a$  is  $5 \log n - 7$ . Lastly, after stretching  $\Gamma_2$ , the height of the final drawing is 4 units taller than  $\Gamma'_a$  (as opposed to 2 in the original version), resulting in  $5 \log n - 3$  height, which satisfies the inductive hypothesis.

We now argue that the width of  $\Gamma$  is linear. Since outer-1-planar graphs have  $O(n)$  number of edges [3], and since our drawings have constant number of bends per edge, it follows that  $\Gamma$  has  $O(n)$  vertical line segments. Note that edges in flat orthogonal box-drawings consist of horizontal and vertical line segments only. After deleting all unnecessary columns, the width of  $\Gamma$  is  $O(n)$ .  $\square$

## 4.4 Conversion to Orthogonal Point-Drawings

In this section we show how to convert a flat orthogonal box-drawing of an outer-1-plane graph to an orthogonal point-drawing by a box-to-orthogonal replacement approach introduced by Tamassia and Tollis [35] and also used by Argyriou et al. [2].

The basic idea is to convert each box (vertex) to an *orthogonal configuration* by a prescribed set of rules. By listing an exhaustive list of “box types”, i.e., all the ways in which incident edge-segments could attach to a box, and associating each box type with its corresponding orthogonal configuration, one can simply make a conversion as shown for one example in Figure 4.10. In the conversion we make sure that the  $x$ -coordinate of each incident edge-segment is unchanged.

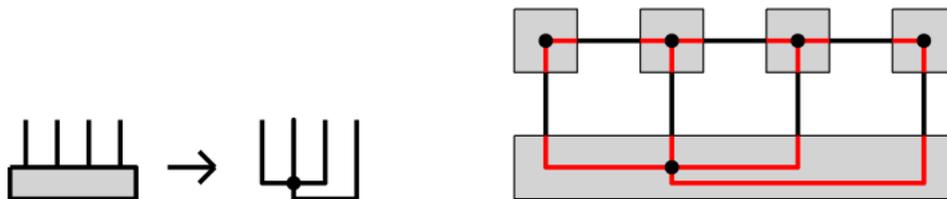


Figure 4.10: An example of replacing a box with the corresponding orthogonal configuration.

We now present the result of a naive conversion method:

**Theorem 19** *Every outer-1-plane graph  $G$  admits an embedding-preserving orthogonal point-drawing in  $O(n \log n)$  area such that any edge has at most 8 bends.*

*Proof.* Let  $\Gamma_{box}$  be a flat orthogonal box-drawing of  $G$  with  $O(n \log n)$  area given by Theorem 18, using any half-skeleton. It is sufficient to ensure that for any edge  $e = (v, u)$  of  $\Gamma_{box}$ , the orthogonal configurations used at  $v$  and  $u$  add no more than 2 bends to  $e$ , and leading to a total of  $4 + 2 + 2 = 8$  bends. See Figure 4.11 for an exhaustive list of replacement rules and notice that no rule adds more than 2 bends to each end of an edge<sup>1</sup>. We now argue that the list of replacement rules presented in Figure 4.11 is exhaustive, up to symmetry.

We assume that all boxes have four incident edge-segments, as otherwise we can simply apply the same conversion rules, ignoring the extra incident edge-segments. Consider the

<sup>1</sup>Argyriou et al. gave a similar list of rules, but we simplified some constructions and added  $(e2)$ ,  $(e3)$ ,  $(f2)$ ,  $(f3)$ , which they did not cover.

two types of boxes: when all the vertical incident edge-segments of a vertex attach to one side of the box, say the top side (*top box*); and when the vertical incident edge-segments attach to both sides of the box (*middle box*). Since boxes are line-segments, only two edges can attach horizontally, so one of these cases must apply if we have four incident edges. The first row of Figure 4.11 outlines all the possible top boxes: (a) 4 vertical incident edge-segments; (b) 3 vertical incident edge-segments; and (c) 2 vertical incident edge-segments (a top box of a degree four vertex cannot have a single vertical incident edge-segment). All the remaining boxes are middle boxes. By considering how many edges attach to each side, but ignoring the relative  $x$ -coordinates of vertical incident edge-segments, we have a total of 4 cases for middle boxes, namely (d), (e), (f) and (g). Lastly, since we keep the  $x$ -coordinates of incident edge-segments unchanged, we consider all relative orderings of incident edge-segments as well, with respect to  $x$ -coordinates. Up to symmetry, these only occur in cases (e1), (e2), (e3) and (f1), (f2), (f3). Note that some cases are omitted due to symmetry; for instance in case (g), the top incident edge-segment could instead be to the left of the bottom incident edge-segment. Cases with coinciding  $x$ -coordinates have also been omitted but can easily be handled (often with fewer bends).

Observe that every replacement rule in Figure 4.11 can be achieved by adding one extra row, and note that we may need to add a new row above a vertex (e.g. case (b)) or below a vertex (e.g. case (f1)). For simplicity, for each vertex  $v$ , add both rows *if necessary*, i.e., to keep the height  $O(\log n)$ , if rows above and below  $v$  are already added, do not add them again. So, adding two extra rows is sufficient to convert all vertices occupying the same row in  $\Gamma_{box}$ . Therefore, the number of rows at most triples, hence the height of the drawing stays  $O(\log n)$ .  $\square$

Note that the area and the total number of bends of the above drawing can be reduced using various techniques, see e.g. [[19], the overview chapter by Eiglsperger et al., Section 6.7]. We will not explore this further to keep the presentation as simple as possible.

To improve the bound to 7 bends per edge, we first prove the following lemma:

**Lemma 7** *Let  $G$  be an outer-1-plane graph and let  $H$  be any half-skeleton of  $skel(G)$ . Then the subgraph composed of missing diagonals of  $H$  is a forest.*

*Proof.* Assume for contradiction that there exists a cycle  $C$  composed of missing diagonals. Cycle  $C$  does not cross itself because for each edge  $e$  of  $C$ , the edge of  $G$  that crosses  $e$  belongs to  $H$  (by definition of a half-skeleton, every missing diagonal is crossed). Therefore,  $C$  defines an interior and an exterior. Further, by outer-1-planarity of  $G$ , every vertex of  $G$  must either be on  $C$  or in the exterior of  $C$ .

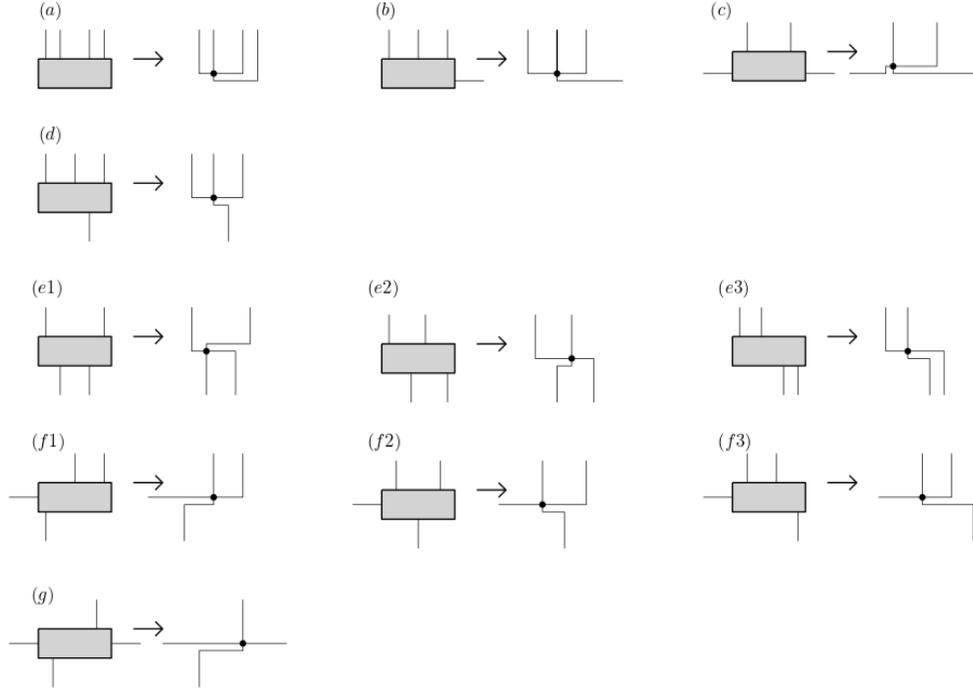


Figure 4.11: Exhaustive list of the box-to-orthogonal conversion rules (up to symmetry). Similar to Argyriou et al. [2].

Enumerate the vertices of  $C$  as  $v_0, v_1, \dots, v_{k-1}$  in clockwise order where  $e_i = (v_i, v_{i+1})$  is an edge for  $0 \leq i \leq k-1 \pmod{k}$ . Let  $e_i^c$  be the edge that crosses  $e_i$  in graph  $G$ , and note that as edge  $e_i^c$  crosses  $e_i$  and enters the interior of  $C$ , its endpoint must be on  $C$ , say it is  $v_{j_i}$ . This is because  $v_{j_i}$  cannot be in the interior of  $C$  since all vertices must be on the outer-face, and it cannot be in the exterior of  $C$  as  $e_i^c$  would have to cross  $C$  for a second time, hence violating 1-planarity of  $G$ .

Now, choose index  $i$  such that  $v_{j_i}$  is closest to edge  $(v_i, v_{i+1})$  on the cycle  $C$ , see Figure 4.12. Assume that  $(j_i - (i+1)) \pmod{k} \leq (i - j_i) \pmod{k}$ , the other case is similar. Note that  $j_i \neq i+1$  since edges with a common endpoint do not cross. So consider the missing diagonal  $e_{i+1} = (v_{i+1}, v_{i+2})$  and its crossing edge  $e_{i+1}^c$ . By 1-planarity,  $e_{i+1}^c$  cannot cross  $e_i^c$ , therefore  $v_{j_{i+1}} \in \{v_{i+3}, \dots, v_{j_i}\}$ . But then  $(j_{i+1} - (i+2)) \pmod{k} < (j_i - (i+1)) \pmod{k}$ , contradicting the choice of  $i$ .  $\square$

We now state the main result of this section:

**Theorem 20** *Every outer-1-plane graph  $G$  admits an embedding-preserving orthogonal*

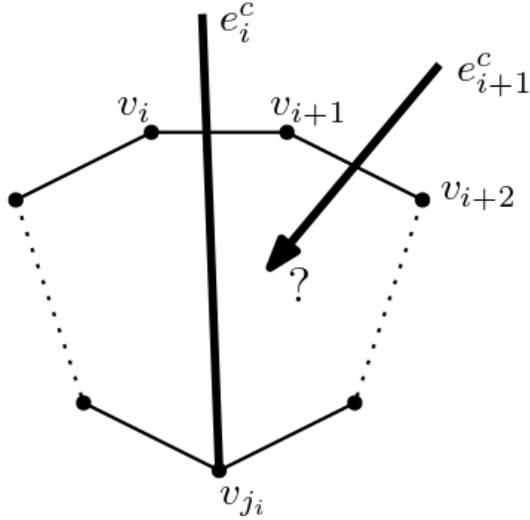


Figure 4.12: Illustration of the proof of Lemma 7.

*drawing in  $O(n \log n)$  area such that any edge has at most 7 bends.*

*Proof.* Let  $H$  be any half-skeleton of  $G$ . Using Theorem 18, obtain a flat orthogonal box-drawing  $\Gamma_{box}$  of  $G$  in  $O(n \log n)$  area such that every missing diagonal of  $H$  has at most 4 bends. By Lemma 7 we know that the subgraph composed of missing diagonals must be a forest. Now, orient the missing diagonals so that every vertex  $v$  has at most one incoming missing diagonal. Call this missing diagonal the *critical incident edge-segment* of  $v$ .

This allows us to “prioritize” the critical incident edge-segment for each vertex  $v$ , and to ensure that it gets at most one extra bend when converting the box of  $v$  to a point. See Figure 4.13 for how to prioritize the (darkened) critical incident edge-segment and let  $\Gamma_{orth}$  be the resulting orthogonal drawing of  $G$ . For example, if  $v$  is a box type (a), then replace the box with either rule (A) or rule (B) depending on whether the critical incident edge-segment is one of the first three edges or the far right edge. Observe that for every box type, every possible incident edge-segment is treated as critical, up to symmetry. Note that some cases are unnecessary (for instance, one could argue that missing diagonals never attach to any box horizontally), but for simplicity of verification all cases are treated.

Since every critical incident edge-segment receives at most one extra bend in the conversion, it follows that every missing diagonal of  $H$  has at most  $4 + 2 + 1 = 7$  bends in  $\Gamma_{orth}$  (recall that the missing diagonals have at most 4 bends in  $\Gamma_{box}$  by Theorem 18). Since all edges of  $H$  have at most 2 bends in  $\Gamma_{box}$ , they will have at most  $2 + 2 + 2 = 6$  bends in

$\Gamma_{orth}$ .

□

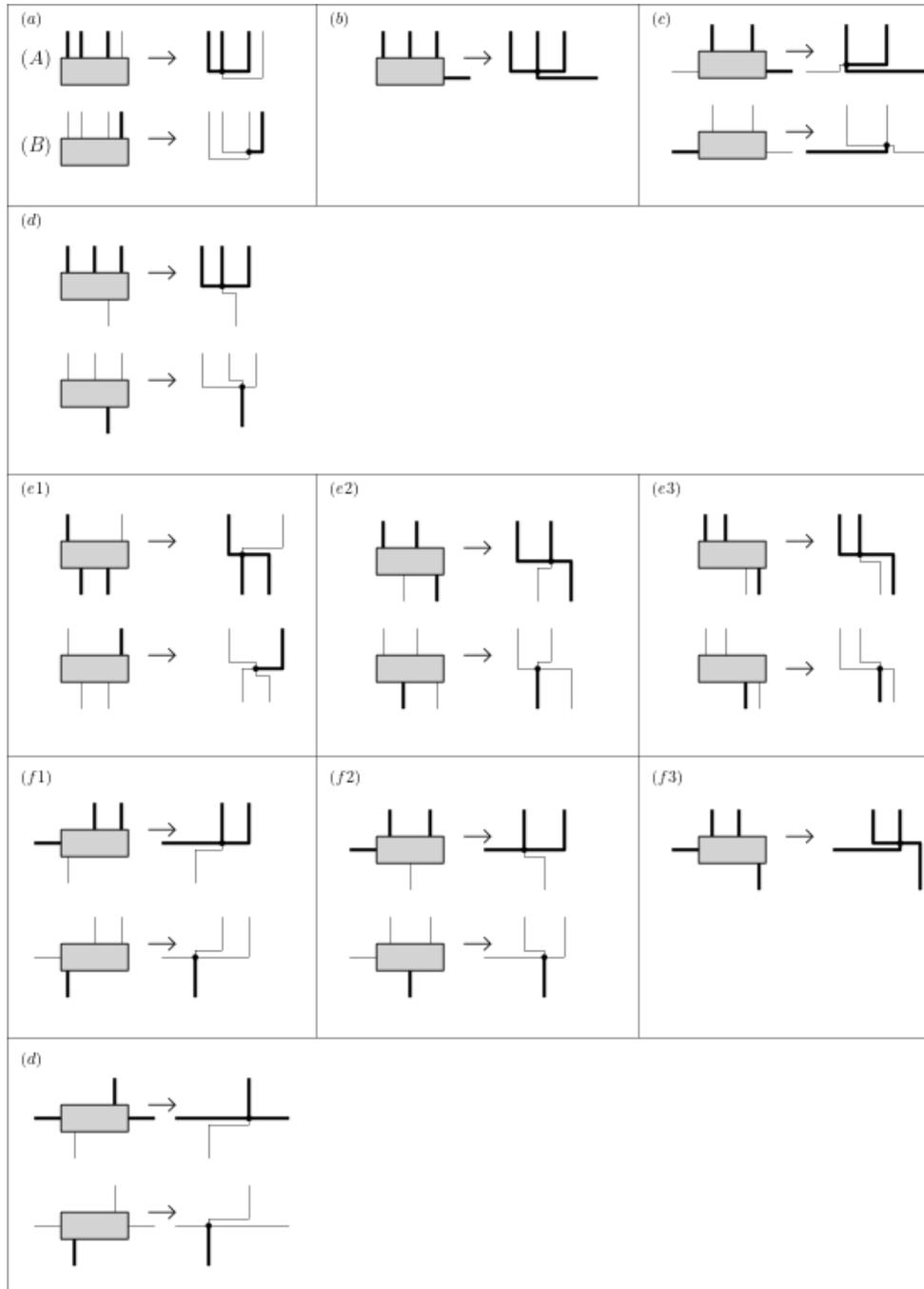


Figure 4.13: Exhaustive list of replacement rules (up to symmetry) which ensures that every critical incident edge-segment gets at most one extra bend. A similar figure was used in [2].

# Chapter 5

## Conclusion and Open Problems

In this thesis we presented several algorithms for producing small straight-line and orthogonal drawings of outer-1-planar graphs. We give a short summary and state a few relevant open problems.

### 5.1 Discussion: Straight-line Drawings

The primary motive for Chapter 3 was an attempt to develop an algorithm which produces embedding-preserving straight-line drawings of outer-1-plane graphs in  $o(n^2)$  area. Although we gave two such algorithms for some interesting sub-classes of o1p graphs, namely o1p graphs of small depth and complete o1p graphs, the general question still remains open:

**Open Problem 1** *Do  $n$ -vertex outer-1-plane graphs admit embedding-preserving straight-line drawings in  $o(n^2)$  area?*

The techniques presented by DiBattista and Frati [15] and Frati et al. [23] for drawing outerplanar graphs do not seem to transfer to a more general class of o1p graphs. By applying their algorithms directly on the half-skeleton, most skeleton faces would be drawn convex, but not strictly convex. The most promising approach in this direction would be to modify the algorithm by Garg and Rusu [24], which uses a similar approach as we did in Section 3.3, but “bends” around subgraphs where depth is large. This remains for future studies.

All existing sub-quadratic area algorithms for outerplanar graphs have width (or height)  $\Omega(n)$  while keeping the other axis small. So, a promising research direction would be to first focus on the subclass of outerplanar graphs, and attempt to produce  $o(n^2)$  drawings so that both the height and the width are sub-linear. Frati conjectured the following:

**Conjecture 1** (Frati [22]) *There exist  $n$ -vertex outerplanar graphs that in any straight-line drawing in which one dimension is  $O(n)$  require  $\omega(\log n)$  in the other dimension.*

Although this question is still unresolved for outerplanar graphs, it may be more accessible for the super-class of o1p graphs. It is interesting that the problem of giving any super-linear lower bound for outerplanar graphs is still open. Such a lower bound is known for series-parallel graphs [21], another super-class of outerplanar graphs.

Further, in this thesis we expanded on the techniques from [15] and obtained drawings of complete o1p graphs in  $O(n^{0.63}) \cdot O(n^{0.63})$  area. Can the area bound be reduced to  $O(n)$ ? Recall that the main reason the resulting area of our approach is super-linear is that we “recovered” the o1p drawing from the dual *binary* tree of its half-skeleton, as opposed to its dual *ternary* tree. The resulting drawing would in fact be linear if complete o1p graphs were defined as the o1p graphs whose  $M(G)$  half-skeleton is a complete binary tree. Still, the problem remains open:

**Open Problem 2** *Is there an algorithm that produces embedding-preserving straight-line drawings of complete  $n$ -vertex o1p graphs in  $O(n)$  area?*

## 5.2 Discussion: Orthogonal Drawings

In Chapter 4, we gave an algorithm that produces an orthogonal point-drawing in  $O(n \log n)$  area of any maximum degree 4 o1p graph with at most 7 bends per edge. Recall that the paper that inspired our result, due to Argyriou et al. [2], showed that any o1p graph admits  $O(n^2)$  area drawing with at most 3 bends per edge. Can we reduce the maximum number of bends per edge from 7 to 3 while keeping the area small? To state it precisely:

**Open Problem 3** *Do  $n$ -vertex o1p graphs with degree at most 4 admit embedding-preserving orthogonal drawings in  $O(n \log n)$  area so that every edge has at most 6 bends?*

# References

- [1] Md. Jawaherul Alam, Franz J. Brandenburg, and Stephen G. Kobourov. Straight-line grid drawings of 3-connected 1-planar graphs. In Stephen K. Wismath and Alexander Wolff, editors, *Graph Drawing - 21st International Symposium, GD 2013, Bordeaux, France, September 23-25, 2013, Proceedings*, volume 8242 of *Lecture Notes in Computer Science*, pages 83–94. Springer, 2013.
- [2] Evmorfia N. Argyriou, Sabine Cornelsen, Henry Förster, Michael Kaufmann, Martin Nöllenburg, Yoshio Okamoto, Chrysanthi N. Raftopoulou, and Alexander Wolff. Orthogonal and smooth orthogonal layouts of 1-planar graphs with low edge complexity. In Therese C. Biedl and Andreas Kerren, editors, *Graph Drawing and Network Visualization - 26th International Symposium, GD 2018, Barcelona, Spain, September 26-28, 2018, Proceedings*, volume 11282 of *Lecture Notes in Computer Science*, pages 509–523. Springer, 2018.
- [3] Christopher Auer, Christian Bachmaier, Franz J. Brandenburg, Andreas Gleißner, Kathrin Hanauer, Daniel Neuwirth, and Josef Reislhuber. Outer 1-planar graphs. *Algorithmica*, 74(4):1293–1320, 2016.
- [4] Giuseppe Di Battista, Peter Eades, Roberto Tamassia, and Ioannis G. Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall PTR, USA, 1st edition, 1998.
- [5] Therese Biedl. Small drawings of outerplanar graphs, series-parallel graphs, and other planar graphs. *Discret. Comput. Geom.*, 45(1):141–160, 2011.
- [6] Therese Biedl. A 4-approximation for the height of drawing 2-connected outer-planar graphs. In Thomas Erlebach and Giuseppe Persiano, editors, *Approximation and Online Algorithms - 10th International Workshop, WAOA 2012, Ljubljana, Slovenia, September 13-14, 2012, Revised Selected Papers*, volume 7846 of *Lecture Notes in Computer Science*, pages 272–285. Springer, 2012.

- [7] Therese Biedl. Drawing outer-1-planar graphs revisited. In *Graph Drawing and Network Visualization (GD'20)*. LNCS. Springer, 2020. Poster with a short abstract. To appear, 2020.
- [8] Marek Chrobak and Goos Kant. Convex grid drawings of 3-connected planar graphs. *International Journal of Computational Geometry and Applications*, 7:211–223, 1997.
- [9] Marek Chrobak and Thomas H. Payne. A linear-time algorithm for drawing a planar graph on a grid. *Inf. Process. Lett.*, 54:241–246, 1995.
- [10] Pierluigi Crescenzi, Giuseppe Di Battista, and Adolfo Piperno. A note on optimal area algorithms for upward drawings of binary trees. *Comput. Geom.*, 2:187–200, 1992.
- [11] Pierluigi Crescenzi, Paolo Penna, and Adolfo Piperno. Linear area upward drawings of AVL trees. *Comput. Geom.*, 9(1-2):25–42, 1998.
- [12] Hubert de Fraysseix, János Pach, and Richard Pollack. Small sets supporting fary embeddings of planar graphs. In Janos Simon, editor, *Proceedings of the 20th Annual ACM Symposium on Theory of Computing, May 2-4, 1988, Chicago, Illinois, USA*, pages 426–433. ACM, 1988.
- [13] Hubert de Fraysseix, János Pach, and Richard Pollack. How to draw a planar graph on a grid. *Combinatorica*, 10:41–51, 1990.
- [14] Hooman Reisi Dehkordi and Peter Eades. Every outer-1-plane graph has a right angle crossing drawing. *Int. J. Comput. Geom. Appl.*, 22(6):543–558, 2012.
- [15] Giuseppe Di Battista and Fabrizio Frati. Small area drawings of outerplanar graphs. *Algorithmica*, 54(1):25–53, 2009.
- [16] Walter Didimo, Giuseppe Liotta, and Fabrizio Montecchiani. A survey on graph drawing beyond planarity. *ACM Comput. Surv.*, 52(1):4:1–4:37, 2019.
- [17] Danny Dolev and Howard Trickey. On linear area embedding of planar graphs. Stanford, USA, 1981, technical report.
- [18] Roger Eggleton. Rectilinear drawings of graphs. *Utilitas Mathematica*, 29:149–172, 1986.
- [19] Markus Eiglsperger, Sándor P. Fekete, and Gunnar W. Klau. Orthogonal graph drawing. In Michael Kaufmann and Dorothea Wagner, editors, *Drawing Graphs, Methods and Models*, volume 2025 of *Lecture Notes in Computer Science*, pages 121–171. Springer, 2001.

- [20] István Fáry. On straight-line representation of planar graphs. *Acta Sci. Math. (Szeged)*, 11:229–233, 1948.
- [21] Fabrizio Frati. Lower bounds on the area requirements of series-parallel graphs. *Discret. Math. Theor. Comput. Sci.*, 12(5):139–174, 2010.
- [22] Fabrizio Frati. Straight-line drawings of outerplanar graphs in  $O(dn \log n)$  area. *Comput. Geom.*, 45(9):524–533, 2012.
- [23] Fabrizio Frati, Maurizio Patrignani, and Vincenzo Roselli. LR-drawings of ordered rooted binary trees and near-linear area drawings of outerplanar graphs. *J. Comput. Syst. Sci.*, 107:28–53, 2020.
- [24] Ashim Garg and Adrian Rusu. Area-efficient planar straight-line drawings of outerplanar graphs. *Discret. Appl. Math.*, 155(9):1116–1140, 2007.
- [25] Emilio Di Giacomo, Giuseppe Liotta, and Fabrizio Montecchiani. Drawing outer 1-planar graphs with few slopes. *J. Graph Algorithms Appl.*, 19(2):707–741, 2015.
- [26] Seok-Hee Hong, Peter Eades, Naoki Katoh, Giuseppe Liotta, Pascal Schweitzer, and Yusuke Suzuki. A linear-time algorithm for testing outer-1-planarity. *Algorithmica*, 72(4):1033–1054, 2015.
- [27] Seok-Hee Hong, Peter Eades, Giuseppe Liotta, and Sheung-Hung Poon. Fáry’s theorem for 1-planar graphs. In Joachim Gudmundsson, Julián Mestre, and Taso Viglas, editors, *Computing and Combinatorics - 18th Annual International Conference, COCOON 2012, Sydney, Australia, August 20-22, 2012. Proceedings*, volume 7434 of *Lecture Notes in Computer Science*, pages 335–346. Springer, 2012.
- [28] John E. Hopcroft and Robert Endre Tarjan. Efficient planarity testing. *J. ACM*, 21(4):549–568, 1974.
- [29] Stephen G. Kobourov, Giuseppe Liotta, and Fabrizio Montecchiani. An annotated bibliography on 1-planarity. *Comput. Sci. Rev.*, 25:49–67, 2017.
- [30] Vladimir P. Korzhik and Bojan Mohar. Minimal obstructions for 1-immersions and hardness of 1-planarity testing. *Journal of Graph Theory*, 72(1):30–71, 2013.
- [31] Sandra L. Mitchell. Linear algorithms to recognize outerplanar and maximal outerplanar graphs. *Information Processing Letters*, 9(5):229 – 232, 1979.

- [32] Adrian Rusu and Andrew Fabian. A straight-line order-preserving binary tree drawing algorithm with linear area and arbitrary aspect ratio. *Comput. Geom.*, 48(3):268 – 294, 2015.
- [33] Walter Schnyder. Embedding planar graphs on the grid. In *Proceedings of the First Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '90, page 138–148, USA, 1990. Society for Industrial and Applied Mathematics.
- [34] Sherman K. Stein. Convex maps. *Proceedings of the American Mathematical Society*, 2:464–466, 1951.
- [35] Roberto Tamassia and Ioannis Tollis. Efficient embedding of planar graphs in linear time. In *Proc. IEEE International Symposium on Circuits and Systems*, pages 495–498, 1987.
- [36] Satoshi Tayu, Kumiko Nomura, and Shuichi Ueno. On the two-dimensional orthogonal drawing of series-parallel graphs. *Discret. Appl. Math.*, 157(8):1885–1895, 2009.
- [37] Carsten Thomassen. Rectilinear drawings of graphs. *Journal of Graph Theory*, 12(3):335–341, 1988.
- [38] Leslie G. Valiant. Universality considerations in VLSI circuits. *IEEE Transactions on Computers*, 30(2):135–140, 1981.
- [39] Klaus Wagner. Bemerkungen zum Vierfarbenproblem. *Jahresber. Deutsch. Math-Verein.*, 46:26–32, 1936.